

# Radio Astronomical Image Formation using Constrained Least Squares and Krylov Subspaces

A. Mouri Sardarabadi<sup>1\*</sup>, Amir Leshem<sup>2</sup>, Alle-Jan van der Veen<sup>1</sup>

## Abstract

Image formation for radio astronomy can be defined as estimating the spatial power distribution of celestial sources over the sky, given an array of antennas. One of the challenges with image formation is that the problem becomes ill-posed as the number of pixels becomes large. The introduction of constraints that incorporate a-priori knowledge is crucial. In this paper we show that in addition to non-negativity, the magnitude of each pixel in an image is also bounded from above. Indeed, the classical “dirty image” is an upper bound, but a much tighter upper bound can be formed from the data using array processing techniques. This formulates image formation as a least squares optimization problem with inequality constraints. We propose to solve this constrained least squares problem using active set techniques, and the steps needed to implement it are described. It is shown that the least squares part of the problem can be efficiently implemented with Krylov subspace based techniques, where the structure of the problem allows massive parallelism and reduced storage needs. The performance of the algorithm is evaluated using simulations.

## Index Terms

Radio astronomy, array signal processing, constrained optimization, Krylov subspace, LSQR, MVDR, image deconvolution

Image formation for radio astronomy can be defined as estimating the spatial power distribution of celestial sources over the sky. The data model (“measurement equation”) is linear in the source powers, and the resulting least squares problem has classically been implemented in two steps: formation of a “dirty image”, followed by a deconvolution step. In this process, an implicit model assumption is made that the number of sources is discrete, and subsequently the number of sources has been replaced by the number of image pixels (assuming each pixel may contain a source).

The deconvolution step becomes ill-conditioned if the number of pixels is large [1]. Alternatively, the directions of sources may be estimated along with their powers, but this is a complex non-linear problem. Classically, this has been implemented as an iterative subtraction technique, wherein source directions are estimated from the dirty image, and their contribution is subtracted from the data. This mixed approach is the essence of the CLEAN method proposed by Högbom [2], which was subsequently refined and extended in several ways, leading to the widely used approaches described in [3], [4].

The conditioning of the image deconvolution step can be improved by incorporating side information such as non-negativity of the image [5], source model structure beyond simple point sources (e.g., shapelets and wavelets [6]), and sparsity or  $\ell_1$  constraints on the image [7], [8]. Beyond these, some fundamental approaches based on parameter estimation techniques have been proposed, such as the Least Squares Minimum Variance Imaging (LS-MVI) [9] and maximum likelihood based techniques [10]. Computational complexity is a concern and this has not been addressed in these approaches.

New radio telescopes such as the Low Frequency Array (LOFAR), the Allen Telescope Array (ATA), Murchison Widefield Array (MWA) and the Long Wavelength Array (LWA) are composed of many stations (each station made up of multiple antennas that are combined using adaptive beamforming), and the increase in number of antennas and stations continues in the design of the square kilometer array (SKA). These instruments have or will have a significantly increased sensitivity and a larger field of view

This research was supported by NWO-TOP 2010, 614.00.005. The research of A. Leshem was supported by the Israeli Science foundation, grant 1240-2009. <sup>1</sup> Department of electrical engineering, Delft University of Technology. <sup>2</sup> Faculty of Engineering, Bar-Ilan University. \*Corresponding author, email: a.mourisardarabadi@tudelft.nl

compared to traditional telescopes, leading to many more sources that need to be taken into account. They also need to process larger bandwidths to reach this sensitivity. Besides the increased requirements on the performance of imaging, the improved spatial resolution leads to an increasing number of pixels in the image, and the development of computationally efficient techniques is critical.

To benefit from the vast literature related to solving least square problems, but also to gain from the non-linear processing offered by standard deconvolution techniques, we propose to reformulate the imaging problem as a parameter estimation problem described by a weighted least squares optimization problem with several constraints. The first is a non-negativity constraint, which would lead to the non-negative least squares algorithm (NNLS) proposed in [5]. But we show that the pixel values are also bounded from above. A coarse upper bound is provided by the classical dirty image, and a much tighter bound is the “minimum variance distortionless response” (MVDR) dirty image that was proposed in the context of radio astronomy in [10].

We propose to solve the resulting constrained least squares problems using an active set approach. This results in a computationally efficient imaging algorithm that is closely related to existing non-linear sequential source estimation techniques such as CLEAN with the benefit of accelerated convergence due to tighter upper bounds on the power distribution over the complete image. Because the constraints are enforced over the entire image, this eliminates the inclusion of negative flux sources and other anomalies that appear in some existing sequential techniques.

To further reduce the computational complexity we show that the data model has a Khatri-Rao structure. This can be exploited to significantly improve the data management and parallelism compared to general implementations of least squares algorithms.

The structure of the paper is as follows. In Sec. I we describe the basic data model, and in Sec. II the image formation problem. A constrained least squares problem is formulated, using various power constraints that take the form of dirty images. The solution of this problem using active set techniques in Sec. III generalizes the classical CLEAN algorithm. In Sec. IV we discuss the efficient implementation of a key step in the active set solution using Krylov subspaces. We end up with some simulated experiments demonstrating the advantages of the proposed technique and conclusions regarding future implementation.

### Notation

A boldface letter such as  $\mathbf{a}$  denotes a column vector, a boldface capital letter such as  $\mathbf{A}$  denotes a matrix. We will frequently use indexed matrices  $\mathbf{A}_k$  and let  $\mathbf{a}_{i,k}$  be the  $i$ th column of  $\mathbf{A}_k$ , whereas  $a_{i,k}$  is the  $i$ th element of the vector  $\mathbf{a}_k$ .  $\mathbf{I}$  is an identity matrix of appropriate size and  $\mathbf{I}_p$  is a  $p \times p$  identity matrix.

$(\cdot)^T$  is the transpose operator,  $(\cdot)^*$  is the complex conjugate operator,  $(\cdot)^H$  is the Hermitian transpose,  $\|\cdot\|_F$  is the Frobenius norm of a matrix,  $\|\cdot\|$  is the two norm of a vector and  $\mathcal{E}\{\cdot\}$  is the expectation operator.

A calligraphic capital letter such as  $\mathcal{X}$  represents a set of indices,

and  $\mathbf{a}_{\mathcal{X}}$  is a column vector constructed by stacking the elements of  $\mathbf{a}$  that belong to  $\mathcal{X}$ . The corresponding indices are stored with the vector as well (similar to the storage of matlab “sparse” vectors).

$\text{vect}(\cdot)$  stacks the columns of the argument matrix to form a vector,  $\text{vectdiag}(\cdot)$  stacks the diagonal elements of the argument matrix to form a vector,  $\text{diag}(\cdot)$  is a diagonal matrix with its diagonal entries from the argument vector (if the argument is a matrix  $\text{diag}(\cdot) = \text{diag}(\text{vectdiag}(\cdot))$ ).

Let  $\otimes$  denote the Kronecker product,  $\circ$  the Khatri-Rao product (column-wise Kronecker product), and  $\odot$  the Hadamard (element-wise) product. The following properties are used throughout the paper (for

matrices and vectors with compatible dimensions):

$$\begin{aligned}
(\mathbf{B}^T \otimes \mathbf{A})\text{vect}(\mathbf{X}) &= \text{vect}(\mathbf{A}\mathbf{X}\mathbf{B}) \\
(\mathbf{B} \otimes \mathbf{A})^H &= (\mathbf{B}^H \otimes \mathbf{A}^H) \\
(\mathbf{B} \otimes \mathbf{A})^{-1} &= (\mathbf{B}^{-1} \otimes \mathbf{A}^{-1}) \\
(\mathbf{B}^T \circ \mathbf{A})\mathbf{x} &= \text{vect}(\mathbf{A}\text{diag}(\mathbf{x})\mathbf{B}) \\
(\mathbf{B}\mathbf{C} \otimes \mathbf{A}\mathbf{D}) &= (\mathbf{B} \otimes \mathbf{A})(\mathbf{C} \otimes \mathbf{D}) \\
(\mathbf{B}\mathbf{C} \circ \mathbf{A}\mathbf{D}) &= (\mathbf{B} \otimes \mathbf{A})(\mathbf{C} \circ \mathbf{D}) \\
(\mathbf{B}^H\mathbf{C} \odot \mathbf{A}^H\mathbf{D}) &= (\mathbf{B} \circ \mathbf{A})^H(\mathbf{C} \circ \mathbf{D}) \\
\text{vectdiag}(\mathbf{A}^H\mathbf{X}\mathbf{A}) &= (\mathbf{A}^* \circ \mathbf{A})^H \text{vect}(\mathbf{X})
\end{aligned}$$

## I. DATA MODEL

We consider an instrument where  $P$  receivers (stations or antennas) are observing the sky. Assuming a discrete point source model, we let  $Q$  denote the number of visible sources. The received signals at the antennas are sampled and subsequently split into narrow sub-bands. For simplicity, we will consider only a single sub-band in the rest of the paper. Although the sources are considered stationary, because of the earth's rotation the apparent position of the celestial sources will change with time. For this reason the data is split into short blocks or “snapshots” of  $N$  samples, where the exact value of  $N$  depends on the resolution of the instrument.

We stack the output of the  $P$  antennas at a single sub-band into a vector  $\mathbf{y}_k[n]$ , where  $n = 1, \dots, N$  denotes the sample index, and  $k = 1, \dots, K$  denotes the snapshot index.

The signals of the  $q$ th source arrive at the array with slight delays for each antenna which depend on the source direction and the earth rotation (the geometric delays), and for sufficiently narrow sub-bands these delays become phase shifts. Let  $\tilde{\mathbf{a}}_{q,k}$  denote this array response vector towards the  $q$ th source at the  $k$ th snapshot. We assume that it is normalized such that  $\tilde{\mathbf{a}}_{q,k}^H \tilde{\mathbf{a}}_{q,k} = 1$ . In this notation, we use a tilde to denote parameters and related matrices that depend on the ‘true’ direction of the sources. However, in most of the paper we will work with parameters that are discretized on a grid, in which case we will drop the tilde. The grid points correspond to the image pixels and do not necessary coincide with the actual positions of the sources.

Assuming an array that is otherwise calibrated, the received antenna signals  $\mathbf{y}_k[n]$  can be modeled as

$$\mathbf{y}_k[n] = \tilde{\mathbf{A}}_k \tilde{\mathbf{x}}[n] + \mathbf{n}_k[n], \quad n = 1, \dots, N \quad (1)$$

where  $\tilde{\mathbf{A}}_k$  is a  $P \times Q$  matrix whose columns are the array response vectors  $\tilde{\mathbf{a}}_{q,k}$ ,  $\tilde{\mathbf{x}}[n]$  is a  $Q \times 1$  vector representing the signals from the sky, and  $\mathbf{n}_k[n]$  is a  $P \times 1$  vector modeling the noise.

From the data, the system estimates covariance matrices (also known as visibilities) of the input vector at each snapshot  $k = 1, \dots, K$ , as

$$\hat{\mathbf{R}}_k = \frac{1}{N} \sum_{n=1}^N \mathbf{y}_k[n] \mathbf{y}_k[n]^H, \quad k = 1, \dots, K. \quad (2)$$

Since the received signals and noise are Gaussian, these covariance matrix estimates form sufficient statistics for the imaging problem [10]. The covariance matrices are given by

$$\mathbf{R}_k = \mathcal{E}\{\mathbf{y}_k \mathbf{y}_k^H\} \quad (3)$$

for which the model is

$$\mathbf{R}_k = \tilde{\mathbf{A}}_k \tilde{\Sigma} \tilde{\mathbf{A}}_k^H + \mathbf{R}_{\mathbf{n},k}, \quad (4)$$

where  $\tilde{\Sigma} = \mathcal{E}\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^H\}$  and  $\mathbf{R}_{\mathbf{n},k} = \mathcal{E}\{\mathbf{n}_k\mathbf{n}_k^H\}$  are the source and noise covariance matrices, respectively. We have assumed that sky sources are stationary, and if we also assume that they are independent, we can model  $\tilde{\Sigma} = \text{diag}(\tilde{\boldsymbol{\sigma}})$  where

$$\tilde{\boldsymbol{\sigma}} = [\tilde{\sigma}_1, \dots, \tilde{\sigma}_Q]^T \quad (5)$$

represents the power of the sources. Vectorizing both sides of (4) we obtain

$$\mathbf{r}_k = (\tilde{\mathbf{A}}_k^* \circ \tilde{\mathbf{A}}_k) \tilde{\boldsymbol{\sigma}} + \mathbf{r}_{\mathbf{n},k} \quad (6)$$

where  $\mathbf{r}_k = \text{vect}(\mathbf{R}_k)$  and  $\mathbf{r}_{\mathbf{n},k} = \text{vect}(\mathbf{R}_{\mathbf{n},k})$ . After stacking the vectorized covariances for all of the snapshots we obtain

$$\mathbf{r} = \tilde{\Psi} \tilde{\boldsymbol{\sigma}} + \mathbf{r}_{\mathbf{n}} \quad (7)$$

where

$$\mathbf{r} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_K \end{bmatrix}, \quad \tilde{\Psi} = \begin{bmatrix} \tilde{\mathbf{A}}_1^* \circ \tilde{\mathbf{A}}_1 \\ \vdots \\ \tilde{\mathbf{A}}_K^* \circ \tilde{\mathbf{A}}_K \end{bmatrix}, \quad \mathbf{r}_{\mathbf{n}} = \begin{bmatrix} \mathbf{r}_{\mathbf{n},1} \\ \vdots \\ \mathbf{r}_{\mathbf{n},K} \end{bmatrix}. \quad (8)$$

Similarly we vectorize and stack the sample covariance matrices as

$$\hat{\mathbf{r}}_k = \text{vect}(\hat{\mathbf{R}}_k), \quad \hat{\mathbf{r}} = \begin{bmatrix} \hat{\mathbf{r}}_1 \\ \vdots \\ \hat{\mathbf{r}}_K \end{bmatrix}. \quad (9)$$

Instead of (7), we can use the independence between the time samples and also write the aggregate data model as

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \dots & \mathbf{0} \\ \vdots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{R}_K \end{bmatrix} = \sum_{q=1}^Q \tilde{\sigma}_q (\mathbf{I}_K \circ \tilde{\mathbf{A}}^q) (\mathbf{I}_K \circ \tilde{\mathbf{A}}^q)^H + \mathbf{R}_{\mathbf{n}}, \quad (10)$$

where

$$\tilde{\mathbf{A}}^q = [\tilde{\mathbf{a}}_{q,1} \quad \dots \quad \tilde{\mathbf{a}}_{q,K}], \quad q = 1, \dots, Q \quad (11)$$

and

$$\mathbf{R}_{\mathbf{n}} = \begin{bmatrix} \mathbf{R}_{\mathbf{n},1} & \dots & \mathbf{0} \\ \vdots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{R}_{\mathbf{n},K} \end{bmatrix}. \quad (12)$$

## II. THE IMAGING PROBLEM

Using the data model (7), the imaging problem is to find the spatial power distribution  $\tilde{\boldsymbol{\sigma}}$  of the sources, along with their directions represented by the matrices  $\tilde{\mathbf{A}}_k$ , from given sample covariance matrices  $\hat{\mathbf{R}}_k$ ,  $k = 1, \dots, K$ . As the source locations are generally unknown, this is a complicated (non-linear) direction-of-arrival estimation problem.

The usual approach in radio astronomy is to define a grid for the image, and to assume that each pixel (grid location) contains a source. In this case the source locations are known, and estimating the source powers is a linear problem, but for high-resolution images the number of sources may be very large. The resulting linear estimation problem is often ill-conditioned unless additional constraints are posed.

### A. Gridded Imaging Model

After defining a grid for the image and assuming that a source exists for each pixel location, let  $I$  denote the total number of sources (pixels),  $\boldsymbol{\sigma}$  an  $I \times 1$  vector containing the source powers, and  $\mathbf{A}_k$  ( $k = 1, \dots, K$ ) the  $P \times I$  array response matrices for these sources. Note that the  $\mathbf{A}_k$  are known, and that  $\boldsymbol{\sigma}$  can be interpreted as a vectorized version of the image to be computed; we dropped the ‘tilde’ in the notation to indicate the difference between the gridded pixel locations and the true (and unknown) source locations. The  $i$ th column of  $\mathbf{A}_k$  is  $\mathbf{a}_{i,k}$ , and similar to  $\tilde{\mathbf{a}}_{q,k}$  we assume that  $\mathbf{a}_{i,k}^H \mathbf{a}_{i,k} = 1$  for  $i = 1, \dots, I$  and  $k = 1, \dots, K$ .

The corresponding data model is now

$$\mathbf{R}_k = \mathbf{A}_k \text{diag}(\boldsymbol{\sigma}) \mathbf{A}_k^H + \mathbf{R}_{n,k}, \quad (13)$$

or in vectorized and stacked form (replacing (7))

$$\mathbf{r} = \boldsymbol{\Psi} \boldsymbol{\sigma} + \mathbf{r}_n, \quad (14)$$

or in blockdiagonal form (replacing (10))

$$\mathbf{R} = \sum_{i=1}^I \sigma_i (\mathbf{I}_K \circ \mathbf{A}^i) (\mathbf{I}_K \circ \mathbf{A}^i)^H + \mathbf{R}_n, \quad (15)$$

where

$$\mathbf{A}_k = [\mathbf{a}_{1,k}, \dots, \mathbf{a}_{I,k}], \quad k = 1, \dots, K \quad (16)$$

$$\boldsymbol{\Psi} = [(\mathbf{A}_1^* \circ \mathbf{A}_1)^T, \dots, (\mathbf{A}_K^* \circ \mathbf{A}_K)^T]^T \quad (17)$$

$$\mathbf{A}^i = [\mathbf{a}_{i,1} \quad \dots \quad \mathbf{a}_{i,K}], \quad i = 1, \dots, I. \quad (18)$$

For a given observation  $\hat{\mathbf{r}}$  in (9), image formation amounts to the estimation of  $\boldsymbol{\sigma}$ . For a sufficiently fine grid,  $\boldsymbol{\sigma}$  approximates the solution of the discrete source model. However, as we will discuss later, working in the image domain leads to a gridding related noise floor. This is solved by fine adaptation of the location of the sources and estimating the true locations in the visibility domain.

### B. Unconstrained Least Squares Images

If we ignore the term  $\mathbf{r}_n$ , then (15) directly leads to Least Squares (LS) and Weighted Least Squares (WLS) estimates of  $\boldsymbol{\sigma}$  [1]. In particular, solving the imaging problem with LS leads to the minimization problem

$$\min_{\boldsymbol{\sigma}} \frac{1}{2K} \|\hat{\mathbf{r}} - \boldsymbol{\Psi} \boldsymbol{\sigma}\|^2. \quad (19)$$

It is straightforward to show that the solution to this problem is given by any  $\boldsymbol{\sigma}$  that satisfies

$$\mathbf{H}_{\text{LS}} \boldsymbol{\sigma} = \hat{\boldsymbol{\sigma}}_{\text{MF}} \quad (20)$$

where we define the “matched filter” (MF, also known as the classical “direct Fourier transform dirty image”) as

$$\hat{\boldsymbol{\sigma}}_{\text{MF}} = \frac{1}{K} \boldsymbol{\Psi}^H \hat{\mathbf{r}} = \frac{1}{K} \sum_k \text{vectdiag}(\mathbf{A}_k^H \hat{\mathbf{R}}_k \mathbf{A}_k), \quad (21)$$

and the deconvolution matrix  $\mathbf{H}_{\text{LS}}$  as

$$\mathbf{H}_{\text{LS}} = \frac{1}{K} \boldsymbol{\Psi}^H \boldsymbol{\Psi} = \frac{1}{K} \sum_k (\mathbf{A}_k^T \mathbf{A}_k^*) \odot (\mathbf{A}_k^H \mathbf{A}_k). \quad (22)$$

Similarly we can define the WLS minimization as

$$\min_{\boldsymbol{\sigma}} \frac{1}{2K} \|(\hat{\mathbf{R}}^{-T/2} \otimes \hat{\mathbf{R}}^{-1/2})(\hat{\mathbf{r}} - \boldsymbol{\Psi}\boldsymbol{\sigma})\|^2, \quad (23)$$

where the weighting assumes Gaussian distributed observations. The weighting improves the statistical properties of the estimates, and  $\hat{\mathbf{R}}$  is used instead of  $\mathbf{R}$  because it is available and gives asymptotically the same optimal results, i.e., convergence to maximum likelihood estimates [11]. The solution to this optimization is similar to the solution to the LS problem and is given by any  $\boldsymbol{\sigma}$  that satisfies

$$\mathbf{H}_{\text{WLS}}\boldsymbol{\sigma} = \hat{\boldsymbol{\sigma}}_{\text{WLS}}, \quad (24)$$

where

$$\hat{\boldsymbol{\sigma}}_{\text{WLS}} = \frac{1}{K} \boldsymbol{\Psi}^H (\hat{\mathbf{R}}^{-T} \otimes \hat{\mathbf{R}}^{-1}) \hat{\mathbf{r}} \quad (25)$$

is the “WLS dirty image” and

$$\mathbf{H}_{\text{WLS}} = \frac{1}{K} \boldsymbol{\Psi}^H (\hat{\mathbf{R}}^{-T} \otimes \hat{\mathbf{R}}^{-1}) \boldsymbol{\Psi}. \quad (26)$$

is the associated deconvolution operator.

A connection to beamforming is obtained as follows. The  $i$ th pixel of the “Matched Filter” dirty image in equation (21) can be written as

$$\hat{\sigma}_{\text{MF},i} = \frac{1}{K} \sum_k \mathbf{a}_{i,k}^H \hat{\mathbf{R}}_k \mathbf{a}_{i,k}$$

and if we replace  $\mathbf{a}_{i,k}/\sqrt{K}$  by a more general “beamformer”  $\mathbf{w}_{i,k}$ , this can be generalized to a more general dirty image

$$\sigma_{\mathbf{w},i} = \sum_k \mathbf{w}_{i,k}^H \hat{\mathbf{R}}_k \mathbf{w}_{i,k}$$

Here,  $\mathbf{w}_{i,k}$  is called a beamformer because we can consider that it acts on the antenna vectors  $\mathbf{y}_k[n]$  as  $z_{i,k} = \mathbf{w}_{i,k}^H \mathbf{y}_k[n]$ , where  $z_{i,k}$  is the output of the (direction-dependent) beamformer, and  $\sigma_{\mathbf{w},i} = \sum_k \mathcal{E}\{|z_{i,k}|^2\}$  is interpreted as the total output power of the beamformer, summed over all snapshots. We will encounter several such beamformers in the rest of the paper.

### C. Preconditioned Weighted Least Squares

If  $\boldsymbol{\Psi}$  has full column rank then  $\mathbf{H}_{\text{LS}}$  and  $\mathbf{H}_{\text{WLS}}$  are non-singular and there exists a unique solution to LS and WLS. For example the solution to (20) becomes

$$\boldsymbol{\sigma} = \mathbf{H}_{\text{LS}}^{-1} \hat{\boldsymbol{\sigma}}_{\text{MF}}. \quad (27)$$

Unfortunately, if the number of pixels is large then  $\mathbf{H}_{\text{LS}}$  and  $\mathbf{H}_{\text{WLS}}$  become ill-conditioned or even singular, so that (20) and (24) have an infinite number of solutions [1]. Generally, we need to improve the conditioning of the deconvolution matrices and to find appropriate regularizations.

One way to improve the conditioning of a matrix is by applying a preconditioner. The most widely used and simplest preconditioner is the Jacobi preconditioner [12] which, for any matrix  $\mathbf{M}$ , is given by  $[\text{diag}(\mathbf{M})]^{-1}$ . Let  $\mathbf{D}_{\text{WLS}} = \text{diag}(\mathbf{H}_{\text{WLS}})$ , then by applying this preconditioner to  $\mathbf{H}_{\text{WLS}}$  we obtain

$$[\mathbf{D}_{\text{WLS}}^{-1} \mathbf{H}_{\text{WLS}}] \boldsymbol{\sigma} = \mathbf{D}_{\text{WLS}}^{-1} \hat{\boldsymbol{\sigma}}_{\text{WLS}}. \quad (28)$$

We take a closer look at  $\mathbf{D}_{\text{WLS}}^{-1} \hat{\boldsymbol{\sigma}}_{\text{WLS}}$  for the case where  $K = 1$ . In this case

$$\begin{aligned} \mathbf{H}_{\text{WLS}} &= (\mathbf{A}_1^* \circ \mathbf{A}_1)^H (\hat{\mathbf{R}}_1^{-T} \otimes \hat{\mathbf{R}}_1^{-1}) (\mathbf{A}_1^* \circ \mathbf{A}_1) \\ &= (\mathbf{A}^T \hat{\mathbf{R}}_1^{-T} \mathbf{A}_1^*) \odot (\mathbf{A}_1^H \hat{\mathbf{R}}_1^{-1} \mathbf{A}_1) \end{aligned}$$

and

$$\mathbf{D}_{\text{WLS}}^{-1} = \begin{bmatrix} \frac{1}{(\mathbf{a}_{1,1}^H \hat{\mathbf{R}}_1^{-1} \mathbf{a}_{1,1})^2} & & \\ & \ddots & \\ & & \frac{1}{(\mathbf{a}_{I,1}^H \hat{\mathbf{R}}_1^{-1} \mathbf{a}_{I,1})^2} \end{bmatrix}.$$

This means that

$$\begin{aligned} \mathbf{D}_{\text{WLS}}^{-1} \hat{\boldsymbol{\sigma}}_{\text{WLS}} &= \mathbf{D}_{\text{WLS}}^{-1} (\hat{\mathbf{R}}_1^{-T} \otimes \hat{\mathbf{R}}_1^{-1}) (\mathbf{A}_1^* \circ \mathbf{A}_1)^H \hat{\mathbf{r}}_1 \\ &= (\hat{\mathbf{R}}_1^{-T} \mathbf{A}_1^* \mathbf{D}_{\text{WLS}}^{-1/2} \circ \hat{\mathbf{R}}_1^{-1} \mathbf{A}_1 \mathbf{D}_{\text{WLS}}^{-1/2})^H \hat{\mathbf{r}}_1 \end{aligned}$$

which is equivalent to a dirty image that is obtained by applying a beamformer of the form

$$\mathbf{w}_i = \frac{1}{\mathbf{a}_{i,1}^H \hat{\mathbf{R}}_1^{-1} \mathbf{a}_{i,1}} \hat{\mathbf{R}}_1^{-1} \mathbf{a}_{i,1} \quad (29)$$

to both sides of  $\hat{\mathbf{R}}_1$  and stacking the results,  $\hat{\sigma}_i = \mathbf{w}_i^H \hat{\mathbf{R}}_1 \mathbf{w}_i$ , of each pixel into a vector. This beamformer is known in array processing as the Minimum Variance Distortionless Response (MVDR) beamformer [13], and the corresponding MVDR dirty image was introduced in the radio astronomy context in [10].

#### D. Bounds on the Image

Another approach to improve the conditioning of a problem is to introduce appropriate constraints on the solution. Typically, image formation algorithms exploit external information regarding the image in order to regularize the ill-posed problem. For example maximum entropy techniques [14], [15] impose a smoothness condition on the image while the CLEAN algorithm [2] exploits a point source model wherein most of the image is empty, and this has recently been connected to sparse optimization techniques [8].

A lower bound on the image is almost trivial: each pixel in the image represents the power coming from a certain direction, hence is non-negative. This leads to a lower bound  $\boldsymbol{\sigma} \geq \mathbf{0}$ . Such a non-negativity constraint has been studied for example in [5], resulting in a non-negative LS (NNLS) problem

$$\begin{aligned} \min_{\boldsymbol{\sigma}} \quad & \frac{1}{2K} \|\hat{\mathbf{r}} - \boldsymbol{\Psi} \boldsymbol{\sigma}\|^2 \\ \text{subject to} \quad & \mathbf{0} \leq \boldsymbol{\sigma} \end{aligned} \quad (30)$$

A second constraint follows if we also know an upper bound  $\gamma$  such that  $\boldsymbol{\sigma} \leq \gamma$ , which will bound the pixel powers from above. We will propose several choices for  $\gamma$ .

Actual dirty images are based on the sample covariance matrix  $\hat{\mathbf{R}}$  and hence they are random variables. By closer inspection of the  $i$ th pixel of the MF dirty image  $\hat{\boldsymbol{\sigma}}_{\text{MF}}$ , we note that its expected value is given by

$$\sigma_{\text{MF},i} = \frac{1}{K} \sum_k \mathbf{a}_{i,k}^H \mathbf{R}_k \mathbf{a}_{i,k}.$$

Using

$$\mathbf{a}_i = \text{vect}(\mathbf{A}^i) = [\mathbf{a}_{i,1}^T \quad \dots \quad \mathbf{a}_{i,K}^T]^T, \quad (31)$$

and the normalization  $\mathbf{a}_{i,k}^H \mathbf{a}_{i,k} = 1$ , we obtain

$$\sigma_{\text{MF},i} = \frac{1}{K} \mathbf{a}_i^H \mathbf{R} \mathbf{a}_i = \sigma_i + \frac{1}{K} \mathbf{a}_i^H \mathbf{R}_r \mathbf{a}_i, \quad (32)$$

where (cf. (15))

$$\mathbf{R}_r = \sum_{j \neq i} \sigma_j (\mathbf{I}_K \circ \mathbf{A}^j) (\mathbf{I}_K \circ \mathbf{A}^j)^H + \mathbf{R}_n \quad (33)$$

is the contribution of all other sources and the noise. Note that  $\mathbf{R}_r$  is positive-(semi)definite. Thus, (32) implies  $\sigma_{\text{MF},i} \geq \sigma_i$  which means that the expected value of the MF dirty image forms an upper bound for the desired image, or

$$\sigma \leq \sigma_{\text{MF}}. \quad (34)$$

Now let us define a beamformer  $\mathbf{w}_{\text{MF},i} = \frac{1}{\sqrt{K}}\mathbf{a}_i$ , then we observe that each pixel in the MF dirty image is the output of this beamformer:

$$\sigma_{\text{MF},i} = \mathbf{w}_{\text{MF},i}^H \mathbf{R} \mathbf{w}_{\text{MF},i}. \quad (35)$$

As indicated in Sec. II-B, we can extend this concept to a more general beamformer  $\mathbf{w}_i$ . The output power of this beamformer, in the direction of the  $i$ th pixel, becomes

$$\sigma_{\mathbf{w},i} = \mathbf{w}_i^H \mathbf{R} \mathbf{w}_i = \sigma_i \mathbf{w}_i^H (\mathbf{I}_K \circ \mathbf{A}^i) (\mathbf{I}_K \circ \mathbf{A}^i)^H \mathbf{w}_i + \mathbf{w}_i^H \mathbf{R}_r \mathbf{w}_i. \quad (36)$$

If we require that

$$\mathbf{w}_i^H (\mathbf{I}_K \circ \mathbf{A}^i) (\mathbf{I}_K \circ \mathbf{A}^i)^H \mathbf{w}_i = 1 \quad (37)$$

we have

$$\sigma_{\mathbf{w},i} = \sigma_i + \mathbf{w}_i^H \mathbf{R}_r \mathbf{w}_i. \quad (38)$$

As before, the fact that  $\mathbf{R}_r$  is positive definite implies that

$$\sigma_i \leq \sigma_{\mathbf{w},i}. \quad (39)$$

We can easily verify that  $\mathbf{w}_{\text{MF},i}$  satisfies (37) and hence  $\sigma_{\text{MF},i}$  is a specific upper bound. A question which arises at this point is: What is the tightest upper bound for  $\sigma_i$  that we can construct using linear beamforming? We can translate this to the following optimization question:

$$\begin{aligned} \sigma_{\text{opt},i} &= \min_{\mathbf{w}_i} \mathbf{w}_i^H \mathbf{R} \mathbf{w}_i \\ \text{s.t. } &\mathbf{w}_i^H (\mathbf{I}_K \circ \mathbf{A}^i) (\mathbf{I}_K \circ \mathbf{A}^i)^H \mathbf{w}_i = 1 \end{aligned} \quad (40)$$

where  $\sigma_{\text{opt},i}$  would be this tightest upper bound. This problem can be solved (Appendix. B): the tightest upper bound is given by

$$\sigma_{\text{opt},i} = \min_k \left( \frac{1}{\mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k}} \right), \quad (41)$$

and the beamformer that achieves this was called the adaptive selective sidelobe canceller (ASSC) in [17]. One problem with using this result in practice is that  $\sigma_{\text{opt},i}$  depends on a single snapshot. This means that there is a variance-bias trade-off when we have a sample covariance matrix  $\hat{\mathbf{R}}$  instead of the true covariance matrix  $\mathbf{R}$ . An analysis of this problem and various solutions for it are discussed in [17].

To reduce the variance we will tolerate an increase of the bound with respect to the tightest, however we would like our result to be tighter than the MF dirty image. For this reason we suggest to find a beamformer that instead of (37) satisfies the slightly different normalization constraint

$$\mathbf{w}_i^H \mathbf{a}_i = \sqrt{K}. \quad (42)$$

We will show that the expected value of the resulting dirty image constitutes a larger upper bound than the ASSC (41), but because the output power of this beamformer depends on more than one snapshot it will have a lower variance than ASSC, so that it is more robust in practice.

With this constraint, the beamforming problem is

$$\begin{aligned} \mathbf{w}_i &= \arg \min_{\mathbf{w}_i} \mathbf{w}_i^H \mathbf{R} \mathbf{w}_i \\ \text{s.t. } &\mathbf{w}_i^H \mathbf{a}_i = \sqrt{K} \end{aligned} \quad (43)$$



which is recognized as the classical minimum variance distortionless response (MVDR) beamforming problem [13]. Thus, the solution is given in closed form as

$$\mathbf{w}_{\text{MVDR},i} = \frac{\sqrt{K}}{\mathbf{a}_i^H \mathbf{R}^{-1} \mathbf{a}_i} \mathbf{R}^{-1} \mathbf{a}_i \quad (44)$$

and the resulting MVDR dirty image is

$$\begin{aligned} \sigma_{\text{MVDR},i} &= \mathbf{w}_{\text{MVDR},i}^H \mathbf{R} \mathbf{w}_{\text{MVDR},i} \\ &= \frac{K \sum_k \mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k}}{\left( \sum_k \mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k} \right)^2} \\ &= \frac{1}{\frac{1}{K} \sum_k \mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k}}. \end{aligned} \quad (45)$$

Interestingly, for  $K = 1$  this is the same image as we obtained earlier by applying a Jacobi preconditioner to the WLS problem. To demonstrate that this image is still an upper bound we show that

$$\alpha := \mathbf{w}_i^H (\mathbf{I}_K \circ \mathbf{A}^i) (\mathbf{I}_K \circ \mathbf{A}^i)^H \mathbf{w}_i \geq 1. \quad (46)$$

Indeed, inserting (44) into this inequality gives

$$\begin{aligned} &K \frac{\mathbf{a}_i^H \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}^i) (\mathbf{I}_K \circ \mathbf{A}^i)^H \mathbf{R}^{-1} \mathbf{a}_i}{(\mathbf{a}_i^H \mathbf{R}^{-1} \mathbf{a}_i)^2} \\ &= K \frac{\sum_k (\mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k})^2}{\left( \sum_k \mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k} \right)^2} \\ &= K \frac{\mathbf{h}^T \mathbf{h}}{\mathbf{h}^T \mathbf{1}_K \mathbf{1}_K^T \mathbf{h}} \geq K \frac{1}{\lambda_{\max}(\mathbf{1}_K \mathbf{1}_K^T)} = 1, \end{aligned} \quad (47)$$

where  $\mathbf{h} = (\mathbf{I}_K \circ \mathbf{A}^i)^H \mathbf{R}^{-1} \mathbf{a}_i$  is a  $K \times 1$  vector with entries  $h_k = \mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k}$  and  $\lambda_{\max}(\cdot)$  is the largest eigenvalue of the argument matrix. Hence, a similar reasoning as in (36) gives

$$\sigma_{\text{MVDR},i} = \alpha \sigma_i + \mathbf{w}_{\text{MVDR},i}^H \mathbf{R}_r \mathbf{w}_{\text{MVDR},i} \geq \sigma_i$$

which is

$$\sigma \leq \sigma_{\text{MVDR}}. \quad (48)$$

Note that  $\mathbf{w}_{\text{MF},i}$  also satisfies the constraint in (43), i.e.  $\mathbf{w}_{\text{MF},i}^H \mathbf{a}_i = \sqrt{K}$ , but does not necessary minimize the output power  $\mathbf{w}_i^H \mathbf{R} \mathbf{w}_i$ , therefore the MVDR dirty image is smaller than the MF dirty image:  $\sigma_{\text{MVDR}} \leq \sigma_{\text{MF}}$ . Thus it is a tighter upper bound. This relation also holds if  $\mathbf{R}$  is replaced by the sample covariance  $\hat{\mathbf{R}}$ .

#### Estimation of the Upper Bound from Noisy Data

The upper bounds (34) and (48) assume that we know the true covariance matrix  $\mathbf{R}$ . However in practice we only measure  $\hat{\mathbf{R}}$  which is subject to statistical fluctuations. Choosing a confidence level of 6 times the standard deviation of the dirty images ensures that the upper bound will hold with probability 99.9%. This leads to an increase of the upper bound by a factor  $1 + \alpha$  where  $\alpha > 0$  is chosen such that

$$\sigma \leq (1 + \alpha) \hat{\sigma}_{\text{MF}}. \quad (49)$$

Similarly, for the MVDR dirty image the constraint based on  $\hat{\mathbf{R}}$  is

$$\sigma \leq (1 + \alpha) \hat{\sigma}_{\text{MVDR}} \quad (50)$$

where

$$\hat{\sigma}_{\text{MVDR},i} = \frac{C}{\frac{1}{K} \sum_k \mathbf{a}_{i,k}^H \hat{\mathbf{R}}_k^{-1} \mathbf{a}_{i,k}} \quad (51)$$

is an unbiased estimate of the MVDR dirty image, and

$$C = \frac{N}{N-p} \quad (52)$$

is a bias correction constant. With some algebra the unbiased estimate can be written in vector form as

$$\hat{\sigma}_{\text{MVDR}} = \mathbf{D}^{-1} \Psi^H (\hat{\mathbf{R}}^{-T} \otimes \hat{\mathbf{R}}^{-1}) \hat{\mathbf{r}}, \quad (53)$$

where

$$\mathbf{D} = \frac{1}{KC} \text{diag}^2 \left( \mathbf{A}^H \hat{\mathbf{R}}^{-1} \mathbf{A} \right), \quad (54)$$

and

$$\begin{aligned} \mathbf{A} &= [\mathbf{A}_1^T \quad \dots \quad \mathbf{A}_K^T]^T \\ &= [\mathbf{a}_1 \quad \dots \quad \mathbf{a}_I]. \end{aligned} \quad (55)$$

The exact choice of  $\alpha$  and  $C$  are discussed in Appendix A.

### E. Constrained Least Squares Imaging

Now that we have lower and upper bounds on the image, we can use these as constraints in the LS imaging problem to provide a regularization. The resulting constrained LS (CLS) imaging problem is

$$\begin{aligned} \min_{\boldsymbol{\sigma}} \quad & \frac{1}{2K} \|\hat{\mathbf{r}} - \Psi \boldsymbol{\sigma}\|^2 \\ \text{s.t.} \quad & \mathbf{0} \leq \boldsymbol{\sigma} \leq \boldsymbol{\gamma} \end{aligned} \quad (56)$$

where  $\boldsymbol{\gamma}$  can be chosen either as  $\boldsymbol{\gamma} = \hat{\boldsymbol{\sigma}}_{\text{MF}}$  for the MF dirty image or  $\boldsymbol{\gamma} = \hat{\boldsymbol{\sigma}}_{\text{MVDR}}$  for the MVDR dirty image.

The improvements to the unconstrained LS problem that were discussed in Sec. II-B are still applicable. The extension to WLS leads to the cost function

$$f_{\text{WLS}}(\boldsymbol{\sigma}) = \frac{1}{2} \|(\hat{\mathbf{R}}^{-T/2} \otimes \hat{\mathbf{R}}^{-1/2}) (\hat{\mathbf{r}} - \Psi \boldsymbol{\sigma})\|^2. \quad (57)$$

The constrained WLS problem is then given by

$$\begin{aligned} \min_{\boldsymbol{\sigma}} \quad & f_{\text{WLS}}(\boldsymbol{\sigma}) \\ \text{s.t.} \quad & \mathbf{0} \leq \boldsymbol{\sigma} \leq \boldsymbol{\gamma}. \end{aligned} \quad (58)$$

We also recommend to include a preconditioner which, as was shown in Sec. II-C, relates the WLS to the MVDR dirty image. However, because of the inequality constraints, (58) does not have a closed form solution and it is solved by an iterative algorithm. In order to have the relation between WLS and MVDR dirty image during the iterations we introduce a change of variable of the form  $\check{\boldsymbol{\sigma}} = \mathbf{D} \boldsymbol{\sigma}$ , where  $\check{\boldsymbol{\sigma}}$  is the new variable for the preconditioned problem and the diagonal matrix  $\mathbf{D}$  is given in (54). The resulting constrained preconditioned WLS (PWLS) optimization problem is

$$\begin{aligned} \check{\boldsymbol{\sigma}} = \arg \min_{\check{\boldsymbol{\sigma}}} \quad & \frac{1}{2} \|(\hat{\mathbf{R}}^{-T/2} \otimes \hat{\mathbf{R}}^{-1/2}) (\hat{\mathbf{r}} - \Psi \mathbf{D}^{-1} \check{\boldsymbol{\sigma}})\|^2 \\ \text{s.t.} \quad & \mathbf{0} \leq \check{\boldsymbol{\sigma}} \leq \mathbf{D} \boldsymbol{\gamma} \end{aligned} \quad (59)$$

and the final image is found by setting  $\boldsymbol{\sigma} = \mathbf{D}^{-1} \check{\boldsymbol{\sigma}}$ . (Here we used that  $\mathbf{D}$  is a positive diagonal matrix so that the transformation to an upper bound for  $\check{\boldsymbol{\sigma}}$  is correct.) Interestingly, the dirty image that follows from the (unconstrained) Weighted Least Squares part of the problem is given by the MVDR image  $\hat{\boldsymbol{\sigma}}_{\text{MVDR}}$  in (53).

### III. CONSTRAINED OPTIMIZATION USING AN ACTIVE SET METHOD

The constrained imaging formulated in the previous section requires the numerical solution of the optimization problems (56) or (59). The problem is classified as a positive definite quadratic program with simple bounds, this is a special case of a convex optimization problem with linear inequality constraints, and we can follow standard approaches to find a solution [18], [19].

For an unconstrained optimization problem, the gradient of the cost function calculated at the solution must vanish. If in an iterative process we are not yet at the optimum, the gradient is used to update the current solution. For constrained optimization, the constraints are usually added to the cost function using (unknown) Lagrange multipliers that need to be estimated along with the solution. At the solution, part of the gradient of the cost function is not zero but related to the nonzero Lagrange multipliers. For inequality constraints, the sign of the Lagrange multipliers plays an important role.

In this Section, we use an approach called the active set method to solve the constrained optimization problem.

#### A. Characterization of the Optimum

Let  $\bar{\sigma}$  be the solution to the optimization problem (56) or (59). An image is called feasible if it satisfies the bounds  $\sigma \geq 0$  and  $-\sigma \geq -\gamma$ . At the optimum, some pixels may satisfy a bound with equality, and these are called the “active” pixels.

We will use the following notation. For any feasible image  $\sigma$ , let

$$\mathcal{L}(\sigma) = \{i \mid \sigma_i = 0\} \quad (60)$$

$$\mathcal{U}(\sigma) = \{i \mid \sigma_i = \gamma_i\} \quad (61)$$

$$\mathcal{A}(\sigma) = \mathcal{L}(\sigma) \cup \mathcal{U}(\sigma) \quad (62)$$

$$\mathcal{F}(\sigma) = \mathcal{I} \setminus \mathcal{A}(\sigma). \quad (63)$$

$\mathcal{I} = \{1, \dots, I\}$  is the set of all pixel indices,  $\mathcal{L}(\sigma)$  is the set where the lower bound is active, i.e., the pixel value is 0.  $\mathcal{U}(\sigma)$  is the set of pixels which attain the upper bound.  $\mathcal{A}(\sigma)$  is the set of all pixels where one of the constraints is active, these are the active pixels. Finally, the free set  $\mathcal{F}(\sigma)$  is the set of pixels  $i$  which have values strictly between 0 and  $\gamma_i$ . Further, for any vector  $\mathbf{v} = [v_i]$ , let  $\mathbf{v}_{\mathcal{F}}$  correspond to the subvector with indices  $i \in \mathcal{F}$ , and similarly define  $\mathbf{v}_{\mathcal{L}}$  and  $\mathbf{v}_{\mathcal{U}}$ . We will write  $\mathbf{v} = \mathbf{v}_{\mathcal{F}} \oplus \mathbf{v}_{\mathcal{L}} \oplus \mathbf{v}_{\mathcal{U}}$ .

Let  $\bar{\sigma}$  be the optimum, and let  $\bar{\mathbf{g}} = \mathbf{g}(\bar{\sigma})$  be the gradient of the cost function at this point. Define the free sets and active sets  $\mathcal{F}, \mathcal{L}, \mathcal{U}$  at  $\bar{\sigma}$ . We can write  $\bar{\mathbf{g}} = \bar{\mathbf{g}}_{\mathcal{F}} \oplus \bar{\mathbf{g}}_{\mathcal{L}} \oplus \bar{\mathbf{g}}_{\mathcal{U}}$ . Associated with the active pixels of  $\bar{\sigma}$  is a vector  $\bar{\boldsymbol{\lambda}} = \bar{\boldsymbol{\lambda}}_{\mathcal{L}} \oplus \bar{\boldsymbol{\lambda}}_{\mathcal{U}}$  of Lagrange multipliers. Optimization theory [18] tells us that the optimum  $\bar{\sigma}$  is characterized by the following conditions:

$$\mathbf{g}_{\mathcal{F}}(\bar{\sigma}) = \mathbf{0} \quad (64)$$

$$\bar{\boldsymbol{\lambda}}_{\mathcal{L}} = \bar{\mathbf{g}}_{\mathcal{L}} \geq \mathbf{0} \quad (65)$$

$$\bar{\boldsymbol{\lambda}}_{\mathcal{U}} = -\bar{\mathbf{g}}_{\mathcal{U}} \geq \mathbf{0}. \quad (66)$$

Thus, the part of the gradient corresponding to the free set is zero, but the part of the gradient corresponding to the active pixels is not necessarily zero. Since we have simple bounds, this part becomes equal to the Lagrange multipliers  $\bar{\boldsymbol{\lambda}}_{\mathcal{L}}$  and  $-\bar{\boldsymbol{\lambda}}_{\mathcal{U}}$  (the negative sign is caused by the condition  $-\sigma_{\mathcal{U}} \geq -\gamma_{\mathcal{U}}$ ). The condition  $\boldsymbol{\lambda} \geq \mathbf{0}$  is crucial: a negative Lagrange multiplier would indicate that there exists a feasible direction of descent  $\mathbf{p}$  for which a small step into that direction,  $\bar{\sigma} + \mu\mathbf{p}$ , has a lower cost and still satisfies the constraints, thus contradicting optimality of  $\bar{\sigma}$  [18].

“Active set” algorithms consider that if the true active set at the solution would be known, the optimization problem with inequality constraints reduces to an optimization with equality constraints,

$$\mathbf{z} = \arg \min_{\sigma} f(\sigma) \quad (67)$$

$$\text{s.t. } \sigma_{\mathcal{L}} = \mathbf{0}, \sigma_{\mathcal{U}} = \gamma_{\mathcal{U}}.$$

Since we can substitute the values of the active pixels into  $\sigma$ , the problem becomes a standard unconstrained LS problem with a reduced dimension: only  $\bar{\sigma}_{\mathcal{F}}$  needs to be estimated. Specifically, for CLS the unconstrained subproblem is formulated as

$$f(\sigma) = \frac{1}{2K} \|\mathbf{b}_{\text{LS}} - \Psi_{\mathcal{F}} \sigma_{\mathcal{F}}\|^2 \quad (68)$$

where

$$\mathbf{b}_{\text{LS}} = \hat{\mathbf{r}} - \Psi_{\mathcal{U}} \sigma_{\mathcal{U}}. \quad (69)$$

Similarly for PWLS we have

$$f(\check{\sigma}) = \frac{1}{2} \left\| \mathbf{b}_{\text{PWLS}} - \left( \hat{\mathbf{R}}^{-T/2} \otimes \hat{\mathbf{R}}^{-1/2} \right) (\Psi \mathbf{D}^{-1})_{\mathcal{F}} \check{\sigma}_{\mathcal{F}} \right\|^2 \quad (70)$$

where

$$\mathbf{b}_{\text{PWLS}} = \left( \hat{\mathbf{R}}^{-T/2} \otimes \hat{\mathbf{R}}^{-1/2} \right) (\hat{\mathbf{r}} - (\Psi \mathbf{D}^{-1})_{\mathcal{U}} \check{\sigma}_{\mathcal{U}}) \quad (71)$$

In both cases, closed form solutions can be found, and we will discuss a suitable Krylov-based algorithm for this in Sec. IV.

Hence the essence of the constrained optimization problem is to find  $\mathcal{L}$ ,  $\mathcal{U}$  and  $\mathcal{F}$ . In the literature algorithms for this are called *active set methods*, and we propose a suitable algorithm in Sec. III-C.

### B. Gradients

We first derive expressions for the gradients required for each of the unconstrained subproblems (68) and (70). Generically, a WLS cost function (as function of a real-valued parameter vector  $\theta$ ) has the form

$$f(\theta)_{\text{WLS}} = \beta \|\mathbf{G}^{1/2} \mathbf{c}(\theta)\|^2 = \beta \mathbf{c}(\theta)^H \mathbf{G} \mathbf{c}(\theta) \quad (72)$$

where  $\mathbf{G}$  is a Hermitian weighting matrix and  $\beta$  is a scalar. The gradient of this function is

$$\mathbf{g}(\theta) = 2\beta \left( \frac{\partial \mathbf{c}}{\partial \theta^T} \right)^H \mathbf{G} \mathbf{c}. \quad (73)$$

For LS we have  $\theta = \sigma$ ,  $\mathbf{c} = \hat{\mathbf{r}} - \Psi \sigma$ ,  $\beta = \frac{1}{2K}$  and  $\mathbf{G} = \mathbf{I}$ . This leads to

$$\begin{aligned} \mathbf{g}_{\text{LS}}(\sigma) &= -\frac{1}{K} \Psi^H (\hat{\mathbf{r}} - \Psi \sigma) \\ &= \mathbf{H}_{\text{LS}} \sigma - \hat{\sigma}_{\text{MF}}. \end{aligned} \quad (74)$$

For PWLS,  $\theta = \check{\sigma}$ ,  $\mathbf{c} = \hat{\mathbf{r}} - \Psi \mathbf{D}^{-1} \check{\sigma}$ ,  $\beta = \frac{1}{2}$  and  $\mathbf{G} = \hat{\mathbf{R}}^{-T} \otimes \hat{\mathbf{R}}^{-1}$ . Substituting into (73) we obtain

$$\begin{aligned} \mathbf{g}_{\text{PWLS}}(\check{\sigma}) &= -\mathbf{D}^{-1} \Psi^H (\hat{\mathbf{R}}^{-T} \otimes \hat{\mathbf{R}}^{-1}) (\hat{\mathbf{r}} - \Psi \mathbf{D}^{-1} \check{\sigma}) \\ &= \mathbf{H}_{\text{PWLS}} \check{\sigma} - \hat{\sigma}_{\text{MVDR}} \end{aligned} \quad (75)$$

where

$$\mathbf{H}_{\text{PWLS}} = \mathbf{D}^{-1} \Psi^H (\hat{\mathbf{R}}^{-T} \otimes \hat{\mathbf{R}}^{-1}) \Psi \mathbf{D}^{-1}, \quad (76)$$

and we used (53).

An interesting observation is that the gradients can be interpreted as residual images obtained by subtracting the dirty image from a convolved model image. This will at a later point allow us to relate the active set method to sequential source removing techniques.

TABLE I: Constrained LS Imaging Using Active Sets

```

1: Initialize: set the initial image  $\sigma^{(0)} = \mathbf{0}$ ,  $j = 0$ , set the free set  $\mathcal{F} = \emptyset$ , and  $\mathcal{L}, \mathcal{U}$  accordingly
2: Set the flag Freegradient-isnotzero := True
3: while Freegradient-isnotzero or  $\lambda_{\min} < 0$  do
4:   if Freegradient-isnotzero then
5:     Let  $\mathbf{z}$  be the solution of the unconstrained subproblem (67)
6:     if  $\mathbf{z}$  is feasible then
7:       Update the image:  $\sigma_{\mathcal{F}}^{(j+1)} = \mathbf{z}$ 
8:       Set Freegradient-isnotzero := False
9:       Compute the “active” part of the gradient and estimate the Lagrange multipliers
10:      Let  $\lambda_{\min}$  be the smallest Lagrange multiplier and  $i_{\min}$  the corresponding pixel index
11:    else
12:      Compute the direction of descent  $\mathbf{p} = \mathbf{z} - \sigma_{\mathcal{F}}^{(j)}$ 
13:      Compute the maximum feasible nonnegative step-size  $\mu_{\max}$  and let  $i$  be the corresponding pixel index that will attain a bound
14:      Update the image:  $\sigma_{\mathcal{F}}^{(j+1)} = \sigma_{\mathcal{F}}^{(j)} + \mu_{\max} \mathbf{p}$ 
15:      Add a constraint: move  $i$  from the free set  $\mathcal{F}$  to  $\mathcal{L}$  or  $\mathcal{U}$ 
16:      Set Freegradient-isnotzero := True
17:    end if
18:    Increase the image index:  $j := j + 1$ 
19:  else
20:    Delete a constraint: move  $i_{\min}$  from  $\mathcal{L}$  or  $\mathcal{U}$  to the free set  $\mathcal{F}$ 
21:    Set Freegradient-isnotzero := True
22:  end if
23: end while

```

### C. Active Set Methods

In this section, we describe the steps needed to find the sets  $\mathcal{L}, \mathcal{U}$  and  $\mathcal{F}$ , and the solution. We follow the template algorithm proposed in [18]. The algorithm is an iterative technique where we gradually improve on an image. Let the image at iteration  $j$  be denoted by  $\sigma^{(j)}$  where  $j = 1, 2, \dots$ , and we always ensure this is a feasible solution (satisfies  $\mathbf{0} \leq \sigma^{(j)} \leq \gamma$ ). The corresponding gradient is the vector  $\mathbf{g} = \mathbf{g}(\sigma^{(j)})$ , and the current estimate of the Lagrange multipliers  $\lambda$  is obtained from  $\mathbf{g}$  using (65), (66). The sets  $\mathcal{L}, \mathcal{U}$  and  $\mathcal{F}$  are current estimates that are not yet necessarily equal to the true sets.

If this image is not yet the true solution, it means that one of the conditions in (64)–(66) is violated. If the gradient corresponding to the free set is not yet zero ( $\mathbf{g}_{\mathcal{F}} \neq \mathbf{0}$ ), then this is remedied by recomputing the image from the essentially unconstrained subproblem (67). It may also happen that some entries of  $\lambda$  are negative. This implies that we do not yet have the correct sets  $\mathcal{L}, \mathcal{U}$  and  $\mathcal{F}$ . Suppose  $\lambda_i < 0$ . The connection of  $\lambda_i$  to the gradient indicates that the cost function can be reduced in that dimension without violating any constraints [18], at the same time making that pixel not active anymore. Thus we remove the  $i$ th pixel from the active set, add it to the free set, and recompute the image with the new equality constraints using (67). As discussed later, a threshold  $\epsilon$  is needed in the test for negativity of  $\lambda_i$  and therefore this step is called the “detection problem”.

Table I summarizes the resulting active set algorithm and describes how the solution  $\mathbf{z}$  to the subproblem is used at each iteration. Some efficiency is obtained by not computing the complete gradient  $\mathbf{g}$  at every iteration, but only the parts corresponding to  $\mathcal{L}, \mathcal{U}$ , when they are needed. For the part corresponding  $\mathcal{F}$ , we use a flag that indicates whether  $\mathbf{g}_{\mathcal{F}}$  is zero or not.

In line 1, the iterative process is initialized. This can be done in many ways. As long as the initial image lies within the feasible region ( $\mathbf{0} \leq \sigma^{(0)} \leq \gamma$ ), the algorithm will converge to a constrained solution. We can simply initialize by  $\sigma^{(0)} = \mathbf{0}$ .

Line 3 is a test for convergence, corresponding to the conditions (64)–(66). The loop is followed while a condition is violated.

If  $\mathbf{g}_{\mathcal{F}}$  is not zero, then in line 5 the unconstrained subproblem (67) is solved. If this solution  $\mathbf{z}$  satisfies the feasibility constraints, then it is kept, the image is updated accordingly, and the gradient is estimated at the new solution (only  $\lambda_{\min} = \min(\lambda)$  is needed, along with the corresponding pixel index).

If  $\mathbf{z}$  is not feasible, then in line 12-16 we try to move into the direction of  $\mathbf{z}$  as far as possible. The direction of descent is  $\mathbf{p} = \mathbf{z} - \boldsymbol{\sigma}_{\mathcal{F}}^{(j)}$ , and the update will be  $\boldsymbol{\sigma}_{\mathcal{F}}^{(j+1)} = \boldsymbol{\sigma}_{\mathcal{F}}^{(j)} + \mu \mathbf{p}$ , where  $\mu$  is a non-negative step size. The  $i$ th pixel will hit a bound if either  $\sigma_i^{(j)} + \mu p_i = 0$  or  $\sigma_i^{(j)} + \mu p_i = \gamma_i$ , i.e., if

$$\mu_i = \max \left( -\frac{\sigma_i^{(j)}}{p_i}, \frac{\gamma_i - \sigma_i^{(j)}}{p_i} \right) \quad (77)$$

(note that  $\mu_i$  is non-negative). Then the maximal feasible step size towards a constraint is given by  $\mu_{\max} = \min(\mu_i)$ , for  $i \in \mathcal{F}$ . The corresponding pixel index is removed from  $\mathcal{F}$  and added to  $\mathcal{L}$  or  $\mathcal{U}$ .

If in line 3 the gradient satisfied  $\mathbf{g}_{\mathcal{F}} = \mathbf{0}$  but a Lagrange multiplier was negative, we delete the corresponding constraint and add this pixel index to the free set (line 20). After this, the loop is entered again with the new constraint sets.

Suppose we initialize the algorithm with  $\boldsymbol{\sigma}^{(0)} = \mathbf{0}$ , then all pixel indices will be in the set  $\mathcal{L}$ , and the free set is empty. During the first iteration  $\boldsymbol{\sigma}_{\mathcal{F}}$  remains empty but the gradient is computed (line 9). Equations (74) and (75) show that it will be equal to the negated dirty image. Thus the minimum of the Lagrange multipliers  $\lambda_{\min}$  will be the current strongest source in the dirty image and it will be added to the free set when the loop is entered again. This shows that the method as described above will lead to a sequential source removal technique similar to CLEAN. In particular, the PWLS cost function (75) relates to LS-MVI [9], which applies CLEAN-like steps to the MVDR dirty image.

In line 3, we try to detect if a pixel should be added to the free set ( $\lambda_{\min} < 0$ ). Note that  $\boldsymbol{\lambda}$  follows from the gradient, (74) or (75), which is a random variable. We should avoid the occurrence of a “false alarm”, because it will lead to overfitting the noise. Therefore, the test should be replaced by  $\lambda_{\min} < -\epsilon$ , where  $\epsilon > 0$  is a suitable detection threshold. Because the gradients are estimated using dirty images, they share the same statistics (the variance of the other component in (74) and (75) is much smaller). To reach a desired false alarm rate, we propose to choose  $\epsilon$  proportional to the standard deviation of the  $i$ th pixel on the corresponding dirty image for the given cost function. (How to estimate the standard deviation of the dirty images and the threshold is discussed in Appendix A). Choosing  $\epsilon$  to be 6 times the standard deviation ensures a false alarm of  $< 0.1\%$  over the complete image.

The use of this statistic improves the detection and hence the estimates greatly, however the correct detection also depends on the quality of the power estimates in the previous iterations. If a strong source is off-grid, the source is usually underestimated, this leads to a biased estimation of the gradient and the Lagrange multipliers, which in turn leads to inclusion of pixels that are not real sources. In the next section we describe one possible solution for this case.

#### D. Strong Off-Grid Sources

The mismatch between  $\boldsymbol{\Psi}$  and the unknown  $\tilde{\boldsymbol{\Psi}}$  results in an underestimation of source powers, which means that the remaining power contribution of that source produces bias and possible artifacts in the image. In order to achieve high dynamic ranges we suggest finding a grid correction for the pixels in the free set  $\mathcal{F}$ .

So far we have not introduced a specific model for the elements in the matrix  $\tilde{\mathbf{A}}_k$ , but in order to be able to correct for these gridding mismatches we assume that the array is at least calibrated for gains such that we can model the columns of this steering matrix as

$$\tilde{\mathbf{a}}_{q,k} = \frac{1}{\sqrt{P}} e^{j \frac{2\pi}{\lambda} \boldsymbol{\Xi}^T \mathbf{Q}_k(L,B) \tilde{\boldsymbol{\beta}}_q} \quad (78)$$

where  $\boldsymbol{\Xi}$  is a  $3 \times P$  matrix containing the position of each receiving element,  $\mathbf{Q}_k$  is a  $3 \times 3$  rotation matrix that accounts for the earth rotations and depends on time and the observer’s longitude  $L$  and latitude  $B$ ,  $\tilde{\boldsymbol{\beta}}_q$  is a  $3 \times 1$  unit vector toward the direction of the source and  $\lambda$  is the wavelength. Let  $\mathbf{a}_{i,k}$  have the

same model as  $\tilde{\mathbf{A}}_{q,k}$  with  $\beta_i$  pointing towards the center of the  $i$ th pixel. When a source is within a pixel but not exactly in the center we can model this mismatch as

$$\begin{aligned}\tilde{\mathbf{a}}_{q,k} &= \frac{1}{\sqrt{P}} e^{j\frac{2\pi}{\lambda} \mathbf{\Xi}^T \mathbf{Q}_k (\beta_i + \delta_i)} \\ &= \mathbf{a}_{i,k} \odot e^{j\frac{2\pi}{\lambda} \mathbf{\Xi}^T \mathbf{Q}_k \delta_i}\end{aligned}$$

where  $\delta_i = \tilde{\beta}_q - \beta_i$  and  $i \in \mathcal{F}$ . Because both  $\beta_i$  and  $\tilde{\beta}_q$  are  $3 \times 1$  unit vectors, each has only two degrees of freedom. This means that we can parameterize the unknowns for the grid correcting problem using coefficients  $\delta_{i,1}$  and  $\delta_{i,2}$ . We will assume that when a source is added to the free set, its actual position is very close to the center of the pixel on which it was detected. This means that  $\delta_{i,1}$  and  $\delta_{i,2}$  are within the pixel's width, denoted by  $W$ , and height, denoted by  $H$ . In this case we can replace (67) by a non-linear constrained optimization,

$$\begin{aligned}\min_{\delta, \sigma} & \frac{1}{2} \|\mathbf{b} - \Psi(\delta)_{\mathcal{F}} \sigma_{\mathcal{F}}\|_2^2 \\ \text{s.t.} & -W/2 < \delta_{i,1} < W/2 \\ & -H/2 < \delta_{i,2} < H/2\end{aligned}\tag{79}$$

where  $\Psi(\delta)_{\mathcal{F}}$  contains only the columns corresponding to the set  $\mathcal{F}$ ,  $\delta_j$  is a vector obtained by stacking  $\delta_{i,j}$  for  $j = 1, 2$  and

$$\mathbf{b} = \hat{\mathbf{r}} - \Psi_{\mathcal{U}} \sigma_{\mathcal{U}}.\tag{80}$$

This problem can also be seen as a direction of arrival (DOA) estimation which is an active research area and out of the scope of this paper. A good review of DOA mismatch correction for MVDR beamformers can be found in [21].

Besides solving (79) instead of (67) in line 5 of the active set method we will also need to update the upper bounds and the standard deviations of the dirty images at the new pixel positions that are used in the other steps (e.g., line 3, 6 and 13), the rest of the steps remain the same. Because we have a good initial guess to where each source in the free set is, we propose a Newton based algorithm to do the correction.

#### IV. IMPLEMENTATION USING KRYLOV SUBSPACE BASED METHODS

From the active set methods described in the previous section, we know that we need to solve (68) or (70) at each iteration. In this section we describe how to achieve this efficiently and without the need of storing the whole convolution matrix in memory.

##### A. Lanczos algorithm and LSQR

When we are solving CLS or PWLS, we need to solve a problem of the form  $\|\mathbf{b} - \mathbf{M}\mathbf{x}\|_2^2$  as the first step of the active set iterations. For example, in (68)  $\mathbf{M} = \Psi_{\mathcal{F}}$ . Note that it does not have to be a square matrix and usually it is ill-conditioned especially if the number of pixels is large. In general we can find a solution for this problem by first computing the singular value decomposition (SVD) of  $\mathbf{M}$  as

$$\mathbf{M} = \mathbf{U}\mathbf{S}\mathbf{V}^H,\tag{81}$$

where  $\mathbf{U}$  and  $\mathbf{V}$  are unitary matrices and  $\mathbf{S}$  is a diagonal matrix with positive singular values. Then the solution  $\mathbf{x}$  to  $\min \|\mathbf{b} - \mathbf{M}\mathbf{x}\|^2$  is found by solving for  $\mathbf{y}$  in

$$\mathbf{S}\mathbf{y} = \mathbf{U}^H \mathbf{b}\tag{82}$$

followed by setting

$$\mathbf{x} = \mathbf{V}\mathbf{y}. \quad (83)$$

Solving the LS problem with this method is expensive in both number of operations and memory usage, especially when the matrices  $\mathbf{U}$  and  $\mathbf{V}$  are not needed after finding the solution. As we will see shortly, looking at another matrix decomposition helps us to reduce these costs. For the rest of this section we use the notation given by [22].

The first step in our approach for solving LS problem is to reduce  $\mathbf{M}$  to a lower bidiagonal form as follows

$$\mathbf{M} = \mathbf{U}\mathbf{B}\mathbf{V}^H, \quad (84)$$

where  $\mathbf{B}$  is a bidiagonal matrix of the form

$$\mathbf{B} = \left[ \begin{array}{ccc|c} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & \beta_r & \alpha_r \\ \hline & & & \mathbf{0} \end{array} \right], \quad (85)$$

with  $r = \text{rank}(\mathbf{M}) = \text{rank}(\mathbf{B})$  and  $\mathbf{U}, \mathbf{V}$  are unitary matrices (different than in (81)). This representation is not unique and without loss of generality we could choose  $\mathbf{U}$  to satisfy

$$\mathbf{U}^H \mathbf{b} = \beta_1 \mathbf{e}_1 \quad (86)$$

where  $\beta_1 = \|\mathbf{b}\|_2$  and  $\mathbf{e}_1$  is a unit norm vector with its first element equal to one.

Using  $\mathbf{B}$ , forward substitution gives the LS solution efficiently by solving  $\mathbf{y}$  in

$$\mathbf{B}\mathbf{y} = \mathbf{U}^H \mathbf{b} = \beta_1 \mathbf{e}_1 \quad (87)$$

followed by

$$\mathbf{x} = \mathbf{V}\mathbf{y}.$$

Using forward substitution we have

$$y_1 = \frac{\beta_1}{\alpha_1} \quad (88)$$

$$\mathbf{x}_1 = \mathbf{v}_1 y_1, \quad (89)$$

followed by the recursion,

$$y_{n+1} = -\frac{\beta_{n+1}}{\alpha_{n+1}} y_n \quad (90)$$

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{v}_{n+1} y_{n+1} \quad (91)$$

for  $n = 1, \dots, M$  where  $M < r$  is the iteration at which  $\|\mathbf{M}^H(\mathbf{M}\mathbf{x}_n - \mathbf{b})\|^2$  vanishes within the desired precision. We can combine the bidiagonalization and solving for  $\mathbf{x}$  and avoid extra storage needed for saving  $\mathbf{B}$ ,  $\mathbf{U}$  and  $\mathbf{V}$ . One such algorithm is based on a Krylov subspace method called the Lanczos algorithm [23]. We first initialize with

$$\beta_1 = \|\mathbf{b}\|_2 \quad (92)$$

$$\mathbf{u}_1 = \frac{\mathbf{b}}{\beta_1} \quad (93)$$

$$\alpha_1 = \|\mathbf{M}^H \mathbf{u}_1\|_2 \quad (94)$$

$$\mathbf{v}_1 = \frac{\mathbf{M}^H \mathbf{u}_1}{\alpha_1}. \quad (95)$$



The iterations are then given by

$$\begin{aligned}
\beta_{n+1} &= \|\mathbf{M}\mathbf{v}_n - \alpha_n \mathbf{u}_n\|_2 \\
\mathbf{u}_{n+1} &= \frac{1}{\beta_{n+1}}(\mathbf{M}\mathbf{v}_n - \alpha_n \mathbf{u}_n) \\
\alpha_{n+1} &= \|\mathbf{M}^H \mathbf{u}_{n+1} - \beta_{n+1} \mathbf{v}_n\|_2 \\
\mathbf{v}_{n+1} &= \frac{1}{\alpha_{n+1}}(\mathbf{M}^H \mathbf{u}_{n+1} - \beta_{n+1} \mathbf{v}_n)
\end{aligned} \tag{96}$$

for  $n = 1, 2, \dots, M$ , where  $\mathbf{u}_n^H \mathbf{u}_n = \mathbf{v}_n^H \mathbf{v}_n = 1$ . This provides us with all the parameters needed to solve the problem.

However because of the finite precision errors, the columns of  $\mathbf{U}$  and  $\mathbf{V}$  found in this way loose their orthogonality as we proceed. In order to prevent this error propagation into the final solution  $\mathbf{x}$ , different algorithms like Conjugate Gradient (CG), MINRES, LSQR, etc. have been proposed. The exact updates for  $\mathbf{x}_n$  and stopping criteria to find  $M$  depends on the choice of algorithm used and therefor is not included in the iterations above.

An overview of Krylov subspace based methods, is given by [24, pp.91]. This study shows that LSQR is a good candidate to solve LS problems when we are dealing with an ill-conditioned and non-square matrix. For this reason we will use LSQR to solve (68) or (70). Because the remaining steps during the LSQR updates are a few scalar operations and do not have large impact on the computational complexity of the algorithm we will not go into the details.(see [22])

In the next section we discuss how to use the structure in  $\mathbf{M}$  to avoid storing the entire matrix in memory and how to parallelize the computations.

### B. Implementation

During the active set iteration we need to solve (68) and (70) where the matrix  $\mathbf{M}$  in LSQR is replaced by  $\Psi_{\mathcal{F}}$  and  $(\mathbf{R}^{-T/2} \otimes \mathbf{R}^{-1/2})(\Psi \mathbf{D}^{-1})_{\mathcal{F}}$  respectively. Because  $\Psi$  has a Khatri-Rao structure and selecting and scaling a subset of columns does not change this,  $\Psi_{\mathcal{F}}$  and  $(\Psi \mathbf{D}^{-1})_{\mathcal{F}}$  also have a Khatri-Rao structure. Here we will show how to use this structure to implement (96) in parallel and with less memory usage.

Note that the only time the matrix  $\mathbf{M}$  enters the algorithm is via the matrix-vector multiplications  $\mathbf{M}\mathbf{v}_n$  and  $\mathbf{M}^H \mathbf{u}_{n+1}$ . As an example we will use  $\mathbf{M} = \Psi_{\mathcal{F}}$  for solving (68). Let  $\mathbf{k}_n = \Psi_{\mathcal{F}} \mathbf{v}_n$ . We partition  $\mathbf{k}_n$  as  $\Psi$  into

$$\mathbf{k}_n = [\mathbf{k}_{1,n}^T \quad \dots \quad \mathbf{k}_{K,n}^T]^T. \tag{97}$$

Using the definition of  $\Psi$  in (17), the operation  $\mathbf{k}_n = \Psi_{\mathcal{F}} \mathbf{v}_n$  could also be performed using

$$\mathbf{K}_{k,n} = \sum_{i \in \mathcal{F}} v_{i,n} \mathbf{a}_{i,k} \mathbf{a}_{i,k}^H. \tag{98}$$

and subsequently setting

$$\mathbf{k}_{k,n} = \text{vect}(\mathbf{K}_{k,n}). \tag{99}$$

This process can be highly parallelized because of the independence between the correlation matrices of each time snapshot. The matrix  $\mathbf{K}_{k,n}$  can then be used to find the updates in (96).

The operation  $\mathbf{M}^H \mathbf{u}$  in (96), is implemented in a similar way. Using the beamforming approach (similar to Sec.II-D), this operation can also be done in parallel for each pixel and each snapshot.

In both cases the calculations can be formulated as correlations and beamforming of parallel data paths which means that efficient hardware implementations are feasible. Also we can consider traditional LS or WLS solutions as a special case when all the pixels belong to the free set which means that those algorithms can also be implemented efficiently in hardware in the same way. Because during the calculations we work with a single beamformer at the time, the matrix  $\Psi$  need not to be pre-calculated and stored in the memory. This makes it possible to apply image formation algorithms for large images when there is a memory shortage.

The computational complexity of the algorithm is dominated by the transformation between the visibility domain and image domain (correlation and beamforming). The dirty image formation and correlation have a complexity of  $O(Kp^2I)$  this means that the worst case complexity of the active set algorithm is  $O(TMKp^2I)$  where  $T$  is the number of active set iterations and  $M$  is the maximum number of Krylov iterations. A direct implementation of CLEAN for solving the imaging problem presented in Sec. II in similar way would have a complexity of  $O(TKp^2I)$ . Hence the proposed algorithm is order  $M$  times more complex, essentially because it recalculates the flux for all the pixels in the free-set while CLEAN only estimates the flux of newly added pixel. In practice many implementations of CLEAN use FFT instead of DFT (matched filter) for calculating the dirty image. Extending the proposed method to use similar techniques is possible and will be presented in future works.

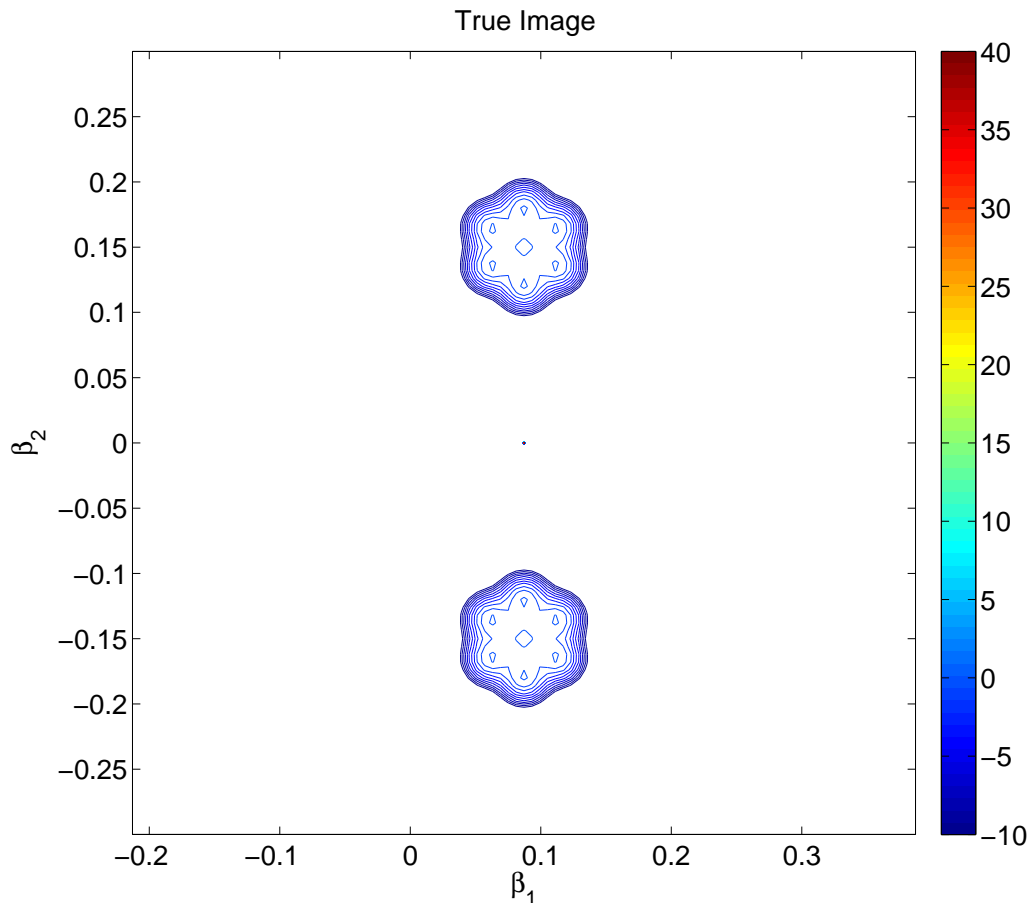


Fig. 1: True source

## V. SIMULATIONS

In this section we will evaluate the performance of the proposed method using simulations. Because the active set algorithm adds a single pixel to the free set at each step, it is important to investigate the effect of this procedure on extended sources and noise. For this purpose we will use a high dynamic range simulated image with a strong point source and two weaker extended sources in the first part of the simulations. In the second part we will make a full sky image using sources from the 3C catalog. We use the following definitions for the coordinate systems. A fixed coordinate system based on the right

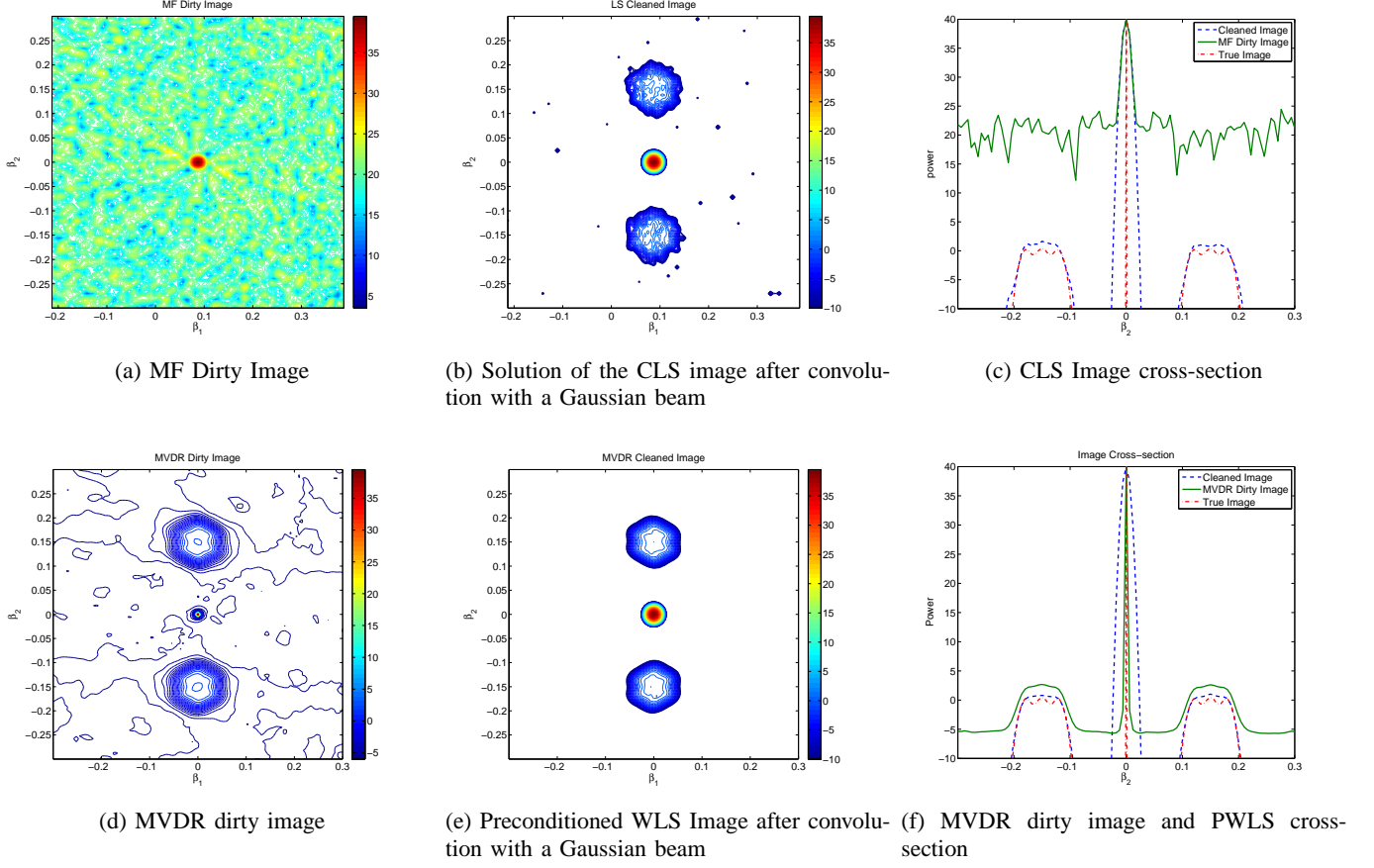


Fig. 2: Extended Source Simulations

ascension ( $\alpha$ ) and declination ( $\delta$ ) of the sources

$$\boldsymbol{\beta} = \begin{bmatrix} \cos(\delta) \cos(\alpha) \\ \cos(\delta) \sin(\alpha) \\ \sin(\delta) \end{bmatrix}$$

The corresponding  $(l, m, n)$  coordinates that take earth rotation into account are given by

$$\mathbf{s} = \mathbf{Q}_k(L, B)\boldsymbol{\beta}.$$

#### A. Extended Sources

An array of 100 dipoles ( $p = 100$ ) with random distribution is used with the frequency range of 58-90 MHz from which we will simulate three equally spaced channels. Each channel has a bandwidth of 195 kHz and is sampled at Nyquist-rate. These specification have been chosen the same as for LOFAR telescope in LBA modes [25]. LOFAR uses 1 second snapshots and we will simulate using only two snapshots, this means that  $K = 2$ . The simulated source is a combination of a strong point source and two extended structures. The extended sources are composed from seven Gaussian shaped sources, one in the middle and 6 on a hexagon around it. Figure 1 shows the simulated image in dB scale. The background noise level that is added is at  $-10$  dB which is also 10 dB below the the extended sources.

Figures 2a and 2d show the matched filter and MVDR dirty images respectively. Figures 2b and 2e show the reconstructed images, after deconvolution and smoothing with a Gaussian clean beam, for the CLS and PWLS deconvolution with MF and MVDR dirty images as upper bounds respectively. A cross section of the images has been illustrated in Figures 2c and 2f. Remarks are:

TABLE II: Simulated Sources from the 3C Catalog

Names	$l$	$m$	Flux
3C 461	-0.30485	0.19131	11000
3C 134	0.59704	-0.02604	66
3C 219	0.63907	0.6598	44
3C 83.1	0.28778	-0.13305	28
3C 75	0.30267	-0.684	23
3C 47	-0.042882	-0.51909	20
3C 399.2	-0.97535	0.20927	19
3C 6.1	-0.070388	0.47098	16
3C 105	0.57458	-0.60492	15
3C 158	0.9017	-0.12339	14
3C 231	0.28956	0.72005	13
3C 303	-0.1511	0.95402	12.5
3C 277.1	0.12621	0.93253	12
3C 320	-0.3597	0.93295	11.5
3C 280.1	0.15171	0.98709	11
3C 454.2	-0.29281	0.31322	10.5
3C 458	-0.61955	-0.56001	10
3C 223.1	0.67364	0.68376	9.5
3C 19	-0.23832	-0.30028	9
3C 437.1	-0.83232	-0.24924	5

- As expected the MVDR dirty image has a much better dynamic range and lower side-lobes;
- Due to a better initial dirty image and upper bound the preconditioned WLS deconvolution gives a better cleaned image. However a trade-off is made between the resolution of the point source and the correct shape of the extended sources when we use the Gaussian beam to smoothen the image.
- The cross sections show the accuracy of the magnitudes. This shows that not only the shape but also the magnitude of the sources are better estimated using PWLS.

### B. Full Sky with 3C Sources

In this part we describe a simulation for making an all sky image. The array setup is the same as before with the same number of channels and snapshots. A background noise level of 0 dB (with respect to 1 Jansky) is added to the sky.

In this simulation we check which sources from the 3C catalog are visible at the simulated date and time. From these we have chosen 20 sources that represent the magnitude distribution on the sky and produce the highest dynamic range available in this catalog. Table II shows the simulated sources with corresponding parameters. The coordinates are the  $(l, m)$  coordinates at the first snapshot. Because the sources are not necessarily on the grid point we have chosen to do the active set deconvolution in combination with grid correction on the free set as described in Sec. III-D.

Figures 3a and 3b show the position and power estimates for the sources that are detected during the deconvolution process. Figure 3c shows the full sky MF dirty image. The contoured version of the reconstructed image with minimum contour 3 dB above the noise level is shown in Figure 3d and the final reconstructed image with the residual added to it is give in Figure 4

Remarks:

- The algorithm stops after adding the correct number of sources based on the detection mechanism we have incorporated in the active set method;
- Because of the grid correction no additional sources are added to compensate for incorrect power estimates on the grids;
- All 20 sources are visible in the final reconstructed image and no visible artifacts are added to image.

## VI. CONCLUSIONS

Based on a parametric model and power constraints, we have formulated image deconvolution as an optimization problem with inequality constraints which we have solved using an active set based method.

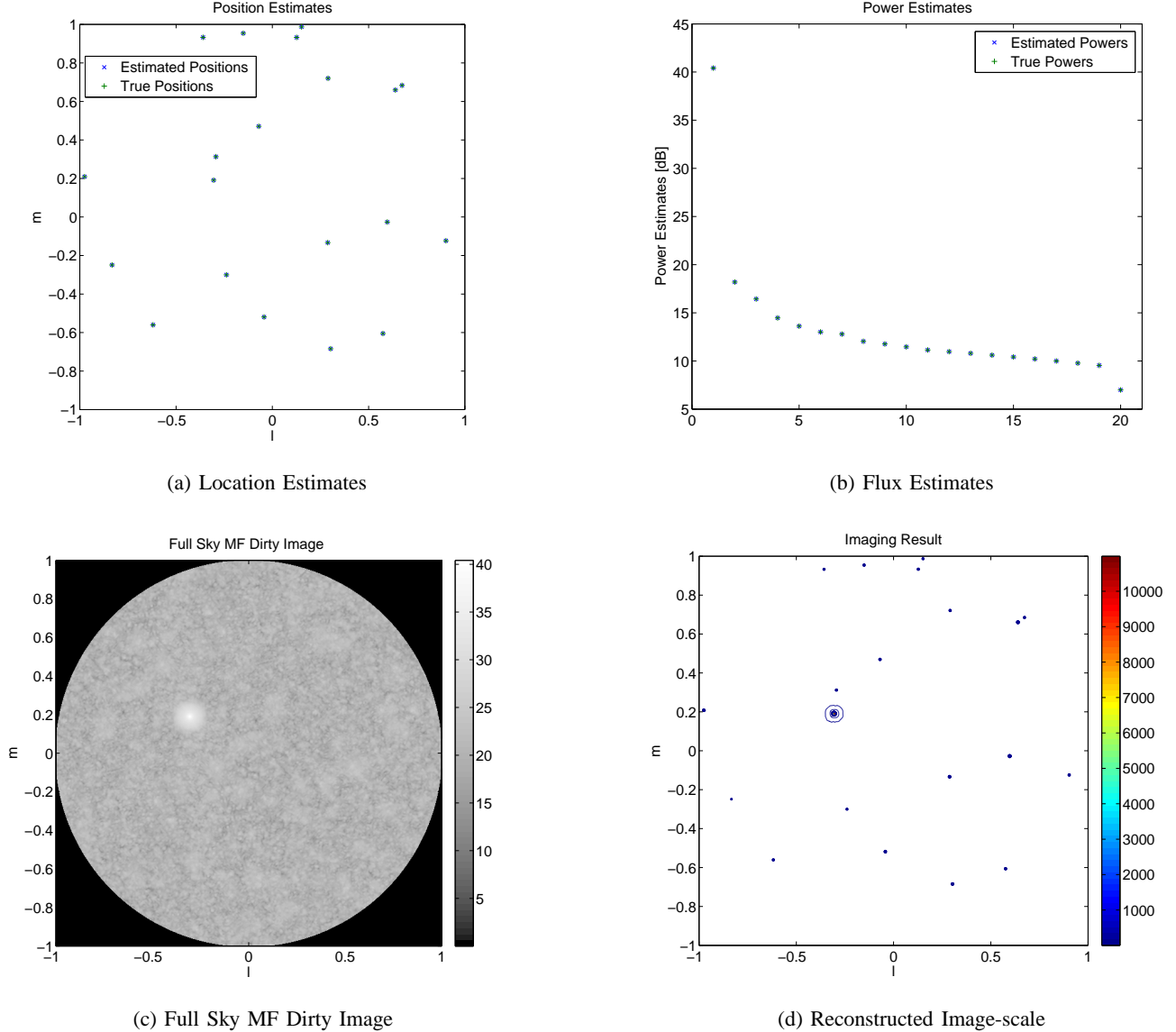


Fig. 3: Point Source Simulations

The relation between the proposed method and sequential source removing techniques is explained. The theoretical background of the active set methods can be used to gain better insight into how the sequential techniques work.

The Khatri-Rao structure of the data model is used in combination with Krylov based techniques to solve the linear systems involved in the deconvolution process with less storage and complexity. We have introduced a preconditioned WLS cost function with a gradient that is related to the MVDR dirty image. Using simulation we have shown that the solution to the preconditioned WLS has improved spatial structure and improved power estimates.

In this paper we have discussed the bidiagonalization and Krylov approach to solve the system of linear equations. The main reason for this is to reduce the storage needed for the deconvolution matrix. It is easy to verify that the active set updates can be translated into rank one updates and downdates of the deconvolution matrix. There are other matrix decompositions like QR decomposition that can take advantage of this fact. Knowing that the Khatri-Rao structure of the matrix does not change by adding or removing columns, it is interesting for future works to investigate whether rank one changes can be

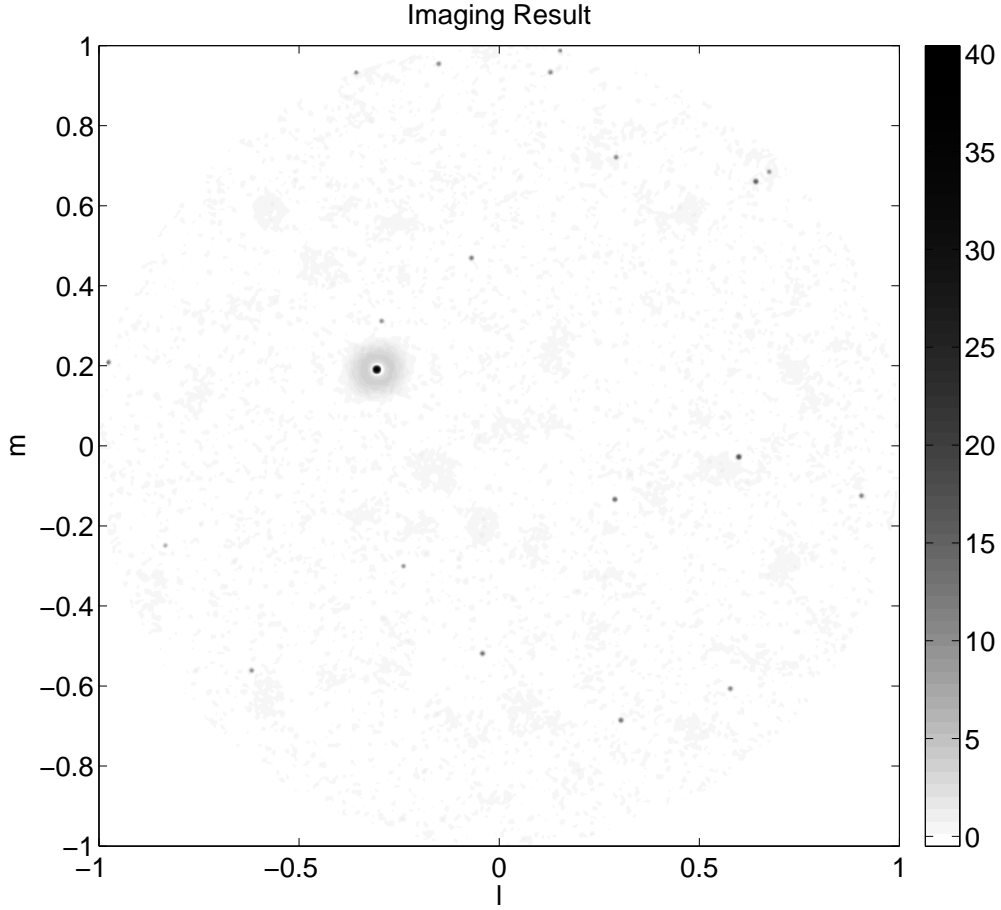


Fig. 4: Reconstructed Image plus the Residual Image

combined with the Krylov based techniques.

#### APPENDIX A UPPER BOUNDS ON IMAGE POWERS

To find the confidence intervals for the dirty images we need to find estimates for the variance of both matched filter and MVDR dirty images. In our problem the sample covariance matrix is obtained by squaring samples from a Gaussian process. This means that  $N\hat{\mathbf{R}} \sim \mathcal{W}_p(\mathbf{R}, N)$  where  $\mathcal{W}_p(\mathbf{R}, N)$  is the Wishart distribution function of order  $p$  with expected value equal to  $\mathbf{R}$  and  $N$  degrees of freedom. For any deterministic vector  $\boldsymbol{\zeta}$ ,

$$N\boldsymbol{\zeta}^H \hat{\mathbf{R}} \boldsymbol{\zeta} \sim \boldsymbol{\zeta}^H \mathbf{R} \boldsymbol{\zeta} \chi^2(N). \quad (100)$$

where  $\chi^2(N)$  is the standard  $\chi^2$  distribution with  $N$  degrees of freedom. In radio astronomical applications  $N$  is usually very large and we can approximate this  $\chi^2$  distribution with a Gaussian such that  $\boldsymbol{\zeta}^H \hat{\mathbf{R}} \boldsymbol{\zeta} \sim \mathcal{N}(\boldsymbol{\zeta}^H \mathbf{R} \boldsymbol{\zeta}, (\boldsymbol{\zeta}^H \mathbf{R} \boldsymbol{\zeta})^2 / N)$ . The variance of the matched filter dirty image is given by

$$\text{Var}(\sigma_{\text{MF},i}) = \frac{1}{NK^2} \sum_k (\mathbf{a}_{i,k}^H \mathbf{R} \mathbf{a}_{i,k})^2$$

Using this result we can find the  $x\%$  confidence interval which results in an increase of the upper bound such that

$$\sigma \leq \hat{\sigma}_{\text{MF}} + \alpha \sqrt{\text{Var}(\hat{\sigma}_{\text{MF}})} \quad (101)$$

where  $\alpha$  is chosen depending on  $x$ . Requiring at most a single false detection on the entire image translate into  $\alpha \approx 6$ .

When we estimate the MVDR dirty image from sample covariance matrices we need to be more careful, mainly because the result is biased and we need to correct for that bias. For each pixel of the MVDR dirty image obtained from sample covariance matrices we have

$$\hat{\sigma}_{\text{MVDR},i} = Kg(Z) = \frac{K}{\sum_k \mathbf{a}_{i,k}^H \hat{\mathbf{R}}_k^{-1} \mathbf{a}_{i,k}} \quad (102)$$

where  $g(Z) = 1/Z$  and  $Z = \sum_k \mathbf{a}_{i,k}^H \hat{\mathbf{R}}_k^{-1} \mathbf{a}_{i,k}$ . Using a perturbation model  $Z = Z_0 + \Delta Z$  and a Taylor approximation we find

$$\begin{aligned} g(Z) &\approx \frac{1}{Z_0} - \frac{1}{Z_0^2} \Delta Z \\ &\approx \frac{1}{Z_0^2} (Z_0 - \Delta Z). \end{aligned} \quad (103)$$

Let  $Z_0 = \mathcal{E}\{Z\}$  then  $\mathcal{E}\{\Delta Z\} = 0$  and  $\mathcal{E}\{g(Z)\} \approx 1/Z_0$ . We would like this estimate to be unbiased which means that we want

$$\mathcal{E}\{g(Z)\} \approx \frac{1}{\sum_k \mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k}} \quad (104)$$

however we have,

$$\begin{aligned} Z_0 &= \sum_k \mathbf{a}_{i,k} \mathcal{E}\{\hat{\mathbf{R}}_k^{-1}\} \mathbf{a}_{i,k} \\ &= \sum_k \mathbf{a}_{i,k}^H \frac{N \mathbf{R}_k^{-1}}{N-p} \mathbf{a}_{i,k} \\ &= \frac{N}{N-p} \sum_k \mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k} \end{aligned} \quad (105)$$

where we have used  $\mathcal{E}\{\hat{\mathbf{R}}^{-1}\} = \frac{N}{N-p} \mathbf{R}^{-1}$  [26]. So in order to remove this bias we need to scale it by a correction factor

$$C = \frac{N}{N-p} \quad (106)$$

and

$$\hat{\sigma}_{\text{MVDR},i} = CKg(Z). \quad (107)$$

Now we need to find an estimate for the variance of the MVDR dirty image. Using (103) we see that the first order approximation of  $\text{Var}(g(Z)) \approx \text{Var}(Z)/Z_0^4$ . We find  $\text{Var}(Z)$  using the independence of each snapshot so we can write

$$\text{Var}(Z) = \sum_k \text{Var}(\mathbf{a}_{i,k}^H \hat{\mathbf{R}}_k^{-1} \mathbf{a}_{i,k}). \quad (108)$$

In order to find  $\text{Var}(\mathbf{a}_{i,k}^H \hat{\mathbf{R}}_k^{-1} \mathbf{a}_{i,k})$  we need to use some properties of the complex inverse Wishart distribution. A matrix has complex inverse Wishart distribution if it's inverse has a complex Wishart distribution [26]. Let us define an invertible matrix  $\mathbf{B}$  as

$$\mathbf{B} = [\mathbf{a}_{i,k} \quad \mathbf{B}_1] \quad (109)$$

then  $\mathbf{X} = (\mathbf{B} \hat{\mathbf{R}}^{-1} \mathbf{B}^H)/N$  has an inverse Wishart distribution because  $\mathbf{X}^{-1} = N(\mathbf{B}^{-H} \hat{\mathbf{R}} \mathbf{B}^{-1})$  has a Wishart distribution. In this case  $\mathbf{X}_{11} = (\mathbf{a}_{i,k}^H \hat{\mathbf{R}}^{-1} \mathbf{a}_{i,k})/N$  also has an inverse Wishart distribution with

less degrees of freedom. The covariance of an inverse Wishart matrix is derived in [26], however because we are dealing only with one element, this results simplifies to

$$\text{Var}(N\mathbf{X}_{11}) = \frac{N^2}{(N-p)^2(N-p-1)} (\mathbf{a}_{i,k}^H \mathbf{R}^{-1} \mathbf{a}_{i,k})^2. \quad (110)$$

The variance of the unbiased MVDR dirty image is thus given by

$$\begin{aligned} \text{Var}(\hat{\sigma}_{\text{MVDR},i}) &= \text{Var}(CKg(Z)) \\ &\approx \frac{K^2}{(N-p-1)} \frac{\sum_k (\mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k})^2}{(\sum_k \mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k})^4}. \end{aligned}$$

Now that we have the variance we can use the same method that we used for MF dirty image to find  $\alpha$  and

$$\sigma \leq \hat{\sigma}_{\text{MVDR}} + \alpha \sqrt{\text{Var}(\hat{\sigma}_{\text{MVDR}})} \quad (111)$$

## APPENDIX B OPTIMUM BEAMFORMER

We have already defined the problem of finding the beamformer for optimum upper bound as

$$\begin{aligned} \mathbf{w}_{i,\text{opt}} &= \arg \min_{\mathbf{w}} \mathbf{w}^H \mathbf{R} \mathbf{w} \\ \text{s.t. } &\mathbf{w}^H (\mathbf{I}_K \circ \mathbf{A}_i) (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{w} = 1 \end{aligned} \quad (112)$$

Following standard optimization techniques we define the Lagrangian and take derivatives with respect to  $\mathbf{w}$  and the Lagrange multiplier  $\mu$  and we find

$$\mathbf{w} = \mu \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{w} \quad (113)$$

$$1 = \mathbf{w}^H (\mathbf{I}_K \circ \mathbf{A}_i) (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{w} \quad (114)$$

Because  $\mathbf{R}$  is full-rank and (114) we can model  $\mathbf{w}$  as

$$\mathbf{w} = \mu \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) \mathbf{x}. \quad (115)$$

Filling back into (113) we have

$$\begin{aligned} &\mu \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) \mathbf{x} \\ &= \mu^2 \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) \mathbf{x} \end{aligned} \quad (116)$$

and

$$\begin{aligned} &(\mathbf{I}_K \circ \mathbf{A}_i) \mathbf{x} \\ &= \mu (\mathbf{I}_K \circ \mathbf{A}_i) (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) \mathbf{x} \end{aligned} \quad (117)$$

multiplying both sides by  $(\mathbf{I}_K \circ \mathbf{A}_i)^H$  we get

$$\mathbf{x} = \mu (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) \mathbf{x}. \quad (118)$$

Doing the same for (114) we have

$$\begin{aligned} &\mu^2 \mathbf{x}^H (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) \mathbf{x} \\ &= 1. \end{aligned} \quad (119)$$

Now we use (118) and we find

$$\mathbf{x}^H \mathbf{x} = 1 \quad (120)$$



which makes finding  $\mathbf{x}$  an eigenvalue problem. By taking a closer look at the matrix  $(\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i)$  we find that this matrix is diagonal

$$\begin{aligned}
 & (\mathbf{I}_K \circ \mathbf{A}_i)^H \mathbf{R}^{-1} (\mathbf{I}_K \circ \mathbf{A}_i) \\
 &= \begin{bmatrix} \mathbf{a}_{i,1}^H \mathbf{R}_1^{-1} \mathbf{a}_{i,1} & 0 & \dots & 0 \\ 0 & \mathbf{a}_{i,2}^H \mathbf{R}_2^{-1} \mathbf{a}_{i,2} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{a}_{i,K}^H \mathbf{R}_K^{-1} \mathbf{a}_{i,K} \end{bmatrix}
 \end{aligned} \tag{121}$$

and hence  $\mathbf{x} = \mathbf{e}_m$  is an elementary vector with all entries equal to zero except for  $m$ th entry which equals unity.  $m$  is the index corresponding to largest eigenvalue,  $\lambda_{\max}$ , and from (118) we have  $\mu = 1/\lambda_{\max}$ . Filling back for  $\mathbf{w}$  we find

$$\mathbf{w}_{i,\text{opt}} = \frac{1}{\mathbf{a}_{i,m}^H \mathbf{R}_m^{-1} \mathbf{a}_{i,m}} \mathbf{R}^{-1} (\mathbf{e}_m \otimes \mathbf{a}_{i,m}) \tag{122}$$

and the output of the beamformer

$$\begin{aligned}
 \sigma_{\text{opt}} &= \mathbf{w}_{i,\text{opt}}^H \mathbf{R} \mathbf{w}_{i,\text{opt}} \\
 &= \frac{\mathbf{a}_{i,m}^H \mathbf{R}_m^{-1} \mathbf{a}_{i,m}}{(\mathbf{a}_{i,m}^H \mathbf{R}_m^{-1} \mathbf{a}_{i,m})^2} \\
 &= \frac{1}{\mathbf{a}_{i,m}^H \mathbf{R}_m^{-1} \mathbf{a}_{i,m}} \\
 &= \min_k \left( \frac{1}{\mathbf{a}_{i,k}^H \mathbf{R}_k^{-1} \mathbf{a}_{i,k}} \right)
 \end{aligned} \tag{123}$$

## REFERENCES

- [1] S. Wijnholds and A.-J. van der Veen, "Fundamental imaging limits of radio telescope arrays," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 5, pp. 613–623, 2008.
- [2] J. A. Högbom, "Aperture synthesis with nonregular distribution of interferometer baselines," *Astron. Astrophys. Suppl.*, vol. 15, pp. 417–426, 1974.
- [3] T. Cornwell, K. Golap, and S. Bhatnagar, "The non-coplanar baselines effect in radio interferometry: The w-projection algorithm," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 5, pp. 647–657, October 2008.
- [4] U. Rau, S. Bhatnagar, M. Voronkov, and T. Cornwell, "Advances in calibration and imaging techniques in radio interferometry," *Proceeding of the IEEE*, vol. 97, pp. 1472–1481, Aug 2009.
- [5] D. S. Briggs, "High fidelity deconvolution of moderately resolved sources," Ph.D. dissertation, The new Mexico Institute of Mining and Technology, Socorro, New Mexico, 1995.
- [6] R. Reid, "Smear fitting: a new image-deconvolution method for interferometric data," *Monthly Notices of the Royal Astronomical Society*, vol. 367, no. 4, pp. 1766–1780, 2006.
- [7] R. Levanda and A. Leshem, "Radio astronomical image formation using sparse reconstruction techniques," *Electrical and Electronics Engineers in Israel, 2008. IEEE 25th Convention of*, pp. 716–720, Dec. 2008.
- [8] Y. Wiaux, L. Jacques, G. Puy, A. Scaife, and P. Vanderghenst, "Compressed sensing imaging techniques for radio interferometry," *Monthly Notices of The Royal Astronomical Society*, Submitted 2009.
- [9] C. Ben-David and A. Leshem, "Parametric high resolution techniques for radio astronomical imaging," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 5, pp. 670–684, Oct. 2008.
- [10] A. Leshem and A. van der Veen, "Radio-astronomical imaging in the presence of strong radio interference," *IEEE Trans. on Information Theory, Special issue on information theoretic imaging*, pp. 1730–1747, August 2000.
- [11] B. Ottersten, P. Stoica, and R. Roy, "Covariance matching estimation techniques for array signal processing applications," *Digital Signal Processing*, vol. 8, no. 3, pp. 185 – 210, 1998. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1051200498903165>
- [12] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. V. der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. Philadelphia, PA: SIAM, 1994.
- [13] J. Capon, "High resolution frequency-wavenumber spectrum analysis," *Proceedings of the IEEE*, pp. 1408–1418, 1969.
- [14] B. Frieden, "Restoring with maximum likelihood and maximum entropy," *Journal of the Optical Society of America*, vol. 62, pp. 511–518, 1972.
- [15] S. Gull and G. Daniell, "Image reconstruction from incomplete and noisy data," *Nature*, vol. 272, pp. 686–690, 1978.
- [16] D. Briggs, "High fidelity deconvolution of moderately resolved sources," Ph.D. dissertation, The New Mexico Institute of Mining and Technology, 1995.

- [17] R. Levanda and A. Leshem, "Adaptive selective sidelobe canceller beamformer with applications to interference mitigation in radio astronomy," *Signal Processing, IEEE Transactions on*, vol. 61, no. 20, pp. 5063–5074, Oct 2013.
- [18] P. E. Gill, W. Murray, and M. H. Wright, *Practical optimization*. London: Academic Press Inc. [Harcourt Brace Jovanovich Publishers], 1981.
- [19] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.
- [20] C. Ben-David and A. Leshem, "Parametric high resolution techniques for radio astronomical imaging," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 5, pp. 670–684, Oct 2008.
- [21] C.-Y. Chen and P. Vaidyanathan, "Quadratically constrained beamforming robust against direction-of-arrival mismatch," *Signal Processing, IEEE Transactions on*, vol. 55, no. 8, pp. 4139–4150, Aug 2007.
- [22] C. C. Paige and M. A. Saunders, "LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares," *ACM Trans. Math. Softw.*, vol. 8, no. 1, pp. 43–71, Mar. 1982.
- [23] G. Golub and W. Kahan, "Calculating the Singular Values and Pseudo-Inverse of a Matrix," *Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis*, vol. 2, no. 2, pp. 205–224, 1965.
- [24] S.-C. T. Choi, "Iterative methods for singular linear equations and least-squares problems," Ph.D. dissertation, Stanford University, 2006.
- [25] M. P. van Haarlem, M. W. Wise, A. W. Gunst *et al.*, "LOFAR: The LOw-Frequency ARray," *A&A*, vol. 556, p. A2, 2013. [Online]. Available: <http://dx.doi.org/10.1051/0004-6361/201220873>
- [26] P. Shaman, "The inverted complex wishart distribution and its application to spectral estimation," *Journal of Multivariate Analysis*, vol. 10, no. 1, pp. 51 – 59, 1980. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0047259X80900810>